Embest ATEB9200
评估板用户手册

Ver 1.0

深圳市英蓓特信息技术有限公司

地址： 深圳市罗湖区太宁路 85 号罗湖科技大厦 509 室（518020）

Tel: +86-755-25631365/25635626 Fax:+86-755-25616057

E-mail: sales@embedinfo.com support@embedinfo.com

http://www.embedinfo.com http://www.embed.com.cn

手册更改记录

更新时间	更新后版本	更新人
2005-06-01	Ver 1.0	Embest L.Han

目　　录

前言......3
第一章 EMBEST ATEB9200 简介......4
1.1 AT91RM9200 芯片简介......4
1.1.1 处理器特点介绍......4
1.1.2 AT91RM9200 芯片资源介绍......4
1.1.3 芯片内部结构框图......5
1.2 EMBEST ATEB9200 简介......6
1.3 EMBEST ATEB9200 评估板资源介绍......6
第二章 EMBEST ATEB9200 硬件介绍......7
2.1 EMBEST ATEB9200 系统框图......7
2.2 功能布局图......8
2.3 硬件端口分配......8
2.4 模块电路介绍......11
2.4.1 启动模式......11
2.4.2 存储器空间映射......12
2.4.3 总线接口扩展接口......13
第三章 EMBEST ATEB9200 开发板安装使用......15
3.1 静电警告......15
3.2 开发板装箱清单......15
3.3 开发板上电默认跳线设置......15
3.4 开发板上电......15
3.5 连接开发板......16
3.5.1 超级终端配置......16
3.5.2 上电前检查......16
3.5.3 开发板上电......16
3.6、光盘内容简单介绍......18
第四章 启动过程分析......19
4.1 芯片启动模式选择......19
4.2 boot 引导过程......19
4.3 Uboot 引导过程......19
4.4 Linux 引导过程......20
第五章 Uboot 安装与应用......21
5.1 Uboot 简单介绍......21
5.2 Uboot 编译......21
5.2.1 环境要求......21
5.2.2 编译步骤......21
5.3 烧写 Uboot 介绍......22
5.3.1 启动 Uboot......22
5.3.2 烧写 Uboot......23
5.4 Uboot 常用命令介绍......26
5.5 Uboot 下的应用程序开发......27
5.5.1 应用程序开发......27

5.5.2 下载应用程序..27
5.5.3 应用程序调试..27
第六章 Linux 安装使用介绍 28
6.1 linux 介绍 28
6.2 linux 编译 28
6.2.1 编译工具安装..28
6.2.2 linux 编译具体步骤..28
6.3 采用 TFTP 方式启动 Linux 29
6.3.1 建立 TFTP 服务器 29
6.3.2 配置 Uboot..30
6.4 采用 DataFlash 启动 Linux 34
第七章 裸机程序开发..36
7.1 使用方法..36
7.2 调试过程示例..36
7.3 程序固化..40
第八章 附录..43
8.1、跳线说明..43
第九章 Embest ATEB9200 系统售后服务及技术支持 44
第十章 免责声明..45

前言

本手册为英蓓特公司 Embest ATEB9200 评估板的用户手册，是 Embest ATEB9200 评估板的配套文档。该手册包含以下章节：

- 第一章 EMBEST ATEB9200 简介
- 第二章 EMBEST ATEB9200 硬件介绍
- 第三章 EMBEST ATEB9200 开发板安装使用
- 第四章 启动过程分析
- 第五章 Uboot 安装与使用
- 第六章 Linux 安装使用介绍
- 第七章 裸机程序开发
- 第八章 附录
- 第九章 Embest ATEB9200 系统售后服务及技术支持
- 第十章 免责声明

用户使用 Embest ATEB9200 评估板与该手册时，还可参考 Embest ATEB9200 评估板的电路原理图(发货光盘中提供)。

因时间仓促，手册中难免存在一些错误，敬请读者谅解，并欢迎指正，谢谢！

深圳市英蓓特信息技术有限公司**©2005**
2005 年版权所有，保留一切权利。
非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本书的部分或全部，并不得以任何形式传播。
Embest®为深圳市英蓓特信息技术有限公司的注册商标，不得仿冒。

Copywrite©2005 by Shenzhen Embest Info&Tech Co., LTD.

All rights reserved.

No part of this document may be reproducted or transmitted in any form or by any means without prior written consent of Embest Info&Tech Co., LTD. Embest® is registered trademarks of Embest Info&Tech Co., LTD.

第一章 EMBEST ATEB9200 简介

1.1 AT91RM9200 芯片简介

1.1.1 处理器特点介绍

AT91RM9200 芯片采用 16-32Bit ARM920T 内核；内部集成有 64K RAM，128K ROM；有外部总线扩展接口（EBI），支持多种类型的外部存储器；内置有主、从的 USB 接口；多达 4 个 USART 接口；内部 SPI 接口；内部 TWI（I2C）接口；内部 EMAC（以太网）接口；内置有 I2S 接口；122 个 I/O 端口，主要针对单芯片高集成、低功耗解决方案设计。

1.1.2 AT91RM9200 芯片资源介绍

主要特性具体描述如下：

- 采用 ARM920T ARM 内核，工作速度达 180MHz，200MIPS，内部分别集成有 16K 的数据和指令 Cache
- 内部存储器：包括 16K 的 SRAM 和 128K 的 ROM
- 外部总线接口：能够支持存储器种类种众多，包括：SDRAM，静态存储器，Burst Flash，CompactFlash，SmartMedia 和 NAND Flash
- 10M/100M 网络接口
- 高性能的系统外围设备：能够产生周期性中断的系统定时器，看门狗；带有报警中断功能的实时时钟；串口调试单元；7 路外部中断输入和 1 路快速中断输入；具有 8 级中断优先级控制的中断控制器；4 路 32-bit 的控制器，外部 I/O 达 122 个；20 通道 DMA
- 2 路 USB HOST 接口
- USB Device 从接口
- MMC/SD 卡接口
- 3 路同步串行控制器
- 4 路 USART，支持 IC 卡、485、红外模块和全功能 MODEM
- 主从模式 SPI 接口
- 3 通道 16 位定时计数器
- 两线接口，支持连接 ATMEL 的 E2PROM
- 处理器正常工作电流为 30.4 毫安，Standby 模式时电流为 3.1 毫安

1.1.3 芯片内部结构框图

AT91RM9200 芯片内部结构框图如下图 1-1。

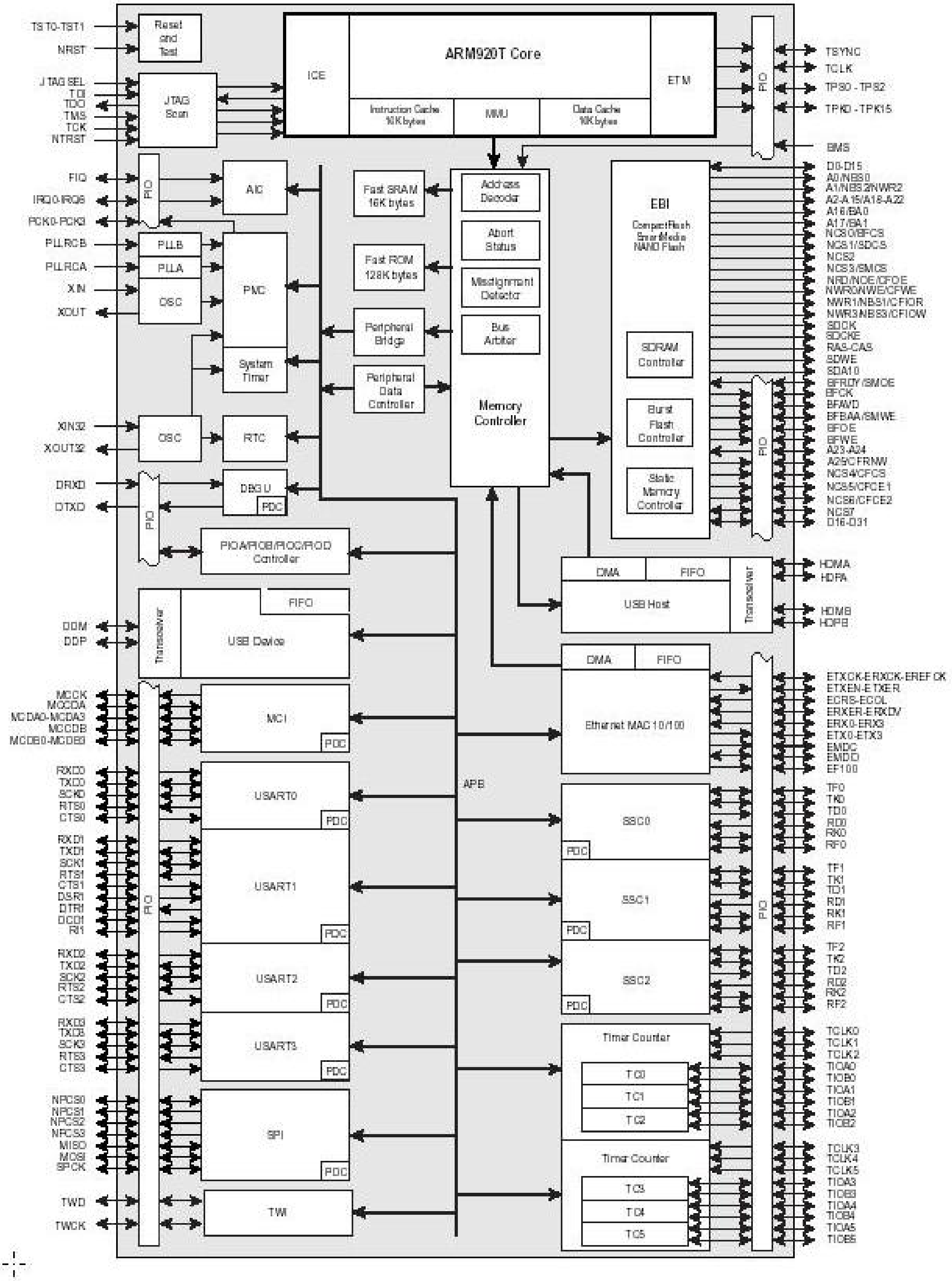

图 1-1

1.2 EMBEST ATEB9200 简介

EMBEST ATEB9200 评估板采用 AT91RM9200 芯片。AT91RM9200 是针对单芯片高集成、低功耗解决方案设计。芯片采用 16-32Bit ARM920T 内核；内部集成有 64K RAM，128K ROM；有外部总线扩展接口（EBI），支持多种类型的外部存储器；内置有主、从的 USB 接口；多达 4 个 USART 接口；内部 SPI 接口；内部 TWI（I2C）接口；内部 EMAC（以太网）接口；内置有 I2S 接口；122 个 I/O 端口。EMBEST ATEB9200 评估板正是基于 AT91RM9200 芯片，利用 AT91RM9200 芯片的丰富的片上资源，然后根据市场新需求的外部扩展少量的模块，整个评估板系统资源非常丰富，设计简洁。

1.3 EMBEST ATEB9200 评估板资源介绍

- AT91RM9200：16-/32-bit ARM920T 内核。最高频率支持 180MHZ
- 2M Flash：采用 Atmel 公司的 AT49BV1614
- 16M SDRAM，最大可支持 64M
- Data Flash 8M（AT45DB642）
- USB：1 个 USB HOST，2 个 USB Device
- 2 个串口：其中一个为简单接口，另一个为全接线接口，可接 RS232 MODEM
- 1 个 IRDA 红外接口
- 1 个 RS485 接口
- 1 个 CAN 接口(可选配 CAN 模块)
- I2C 总线读写 E2PROM
- 1 个 10/100M 以太网接口模块
- AC97 模块
- SD/MMC 卡接口
- CF 卡接口
- SM 卡接口
- I/O 端口：通过插座全部扩出来，以外扩子板等
- ETM 调试接口，可以采用高级开发调试软件进行开发
- 20 针标准 JTAG 接口

第二章 EMBEST ATEB9200 硬件介绍

2.1 EMBEST ATEB9200 系统框图

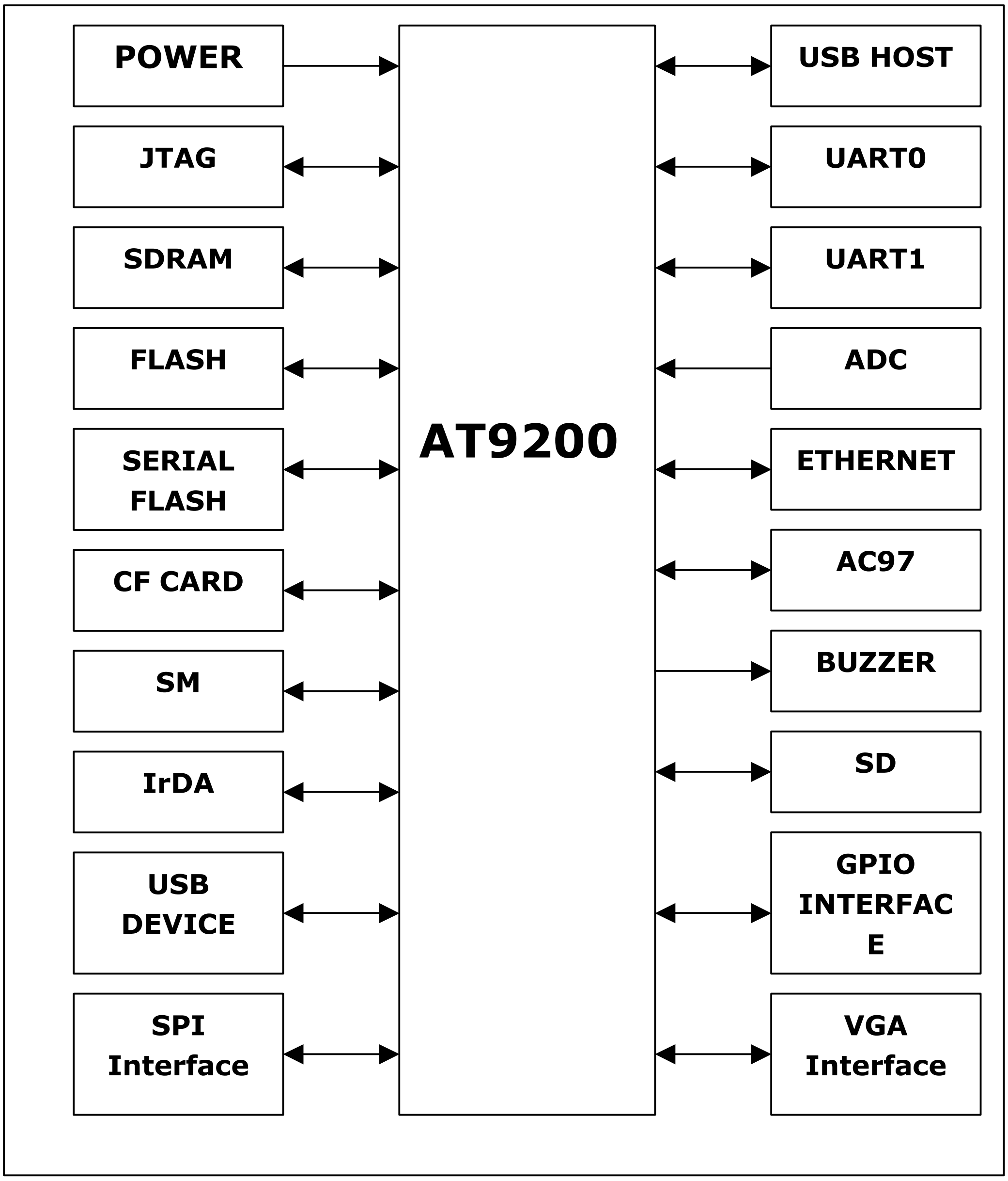

图 2-1

EMBEST ATEB9200 评估板包括：AT91RM9200 处理器，电源模块，JTAG 调试接口，外部 SDRAM，外部 FLASH，外部串行 DATA FLASH，CF 卡接口，SM 接口，IrDA 模块，USB Device 模块，USB Host 模块，串口 UART0，串口 UART1，ETHERNET 模块，AC97 模块，SD 接口，BUZZER，GPIO 外扩接口等。

2.2 功能布局图

整个 ATEB9200 开发板的器件布局图如下图 2-2。

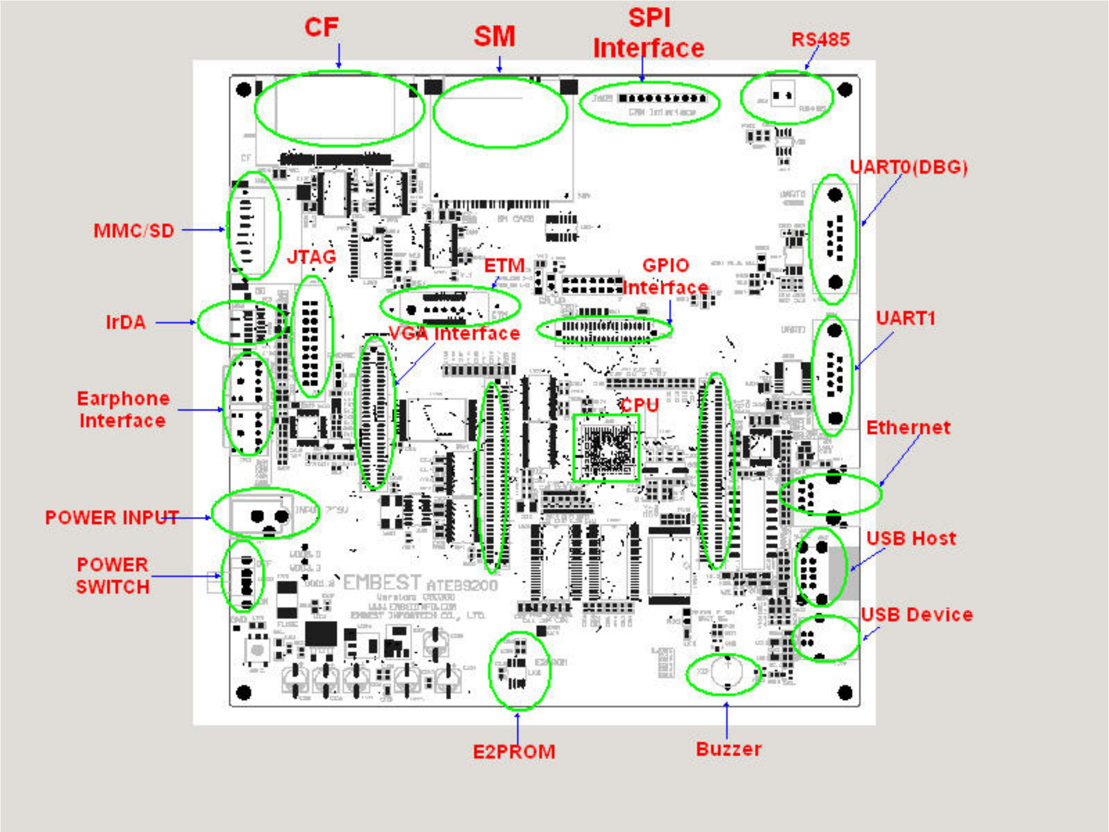

图 2-2

2.3 硬件端口分配

Embest ATEB9200 评估板的硬件端口分配如下：

模块	端口名称	管脚号	输入输出	作用	备注
I2C	PA26			SCK	
	PA25			SDA	
SDRAM	A2~A14			地址总线	
	D0~D15			数据总线	
	NWR1	T7			
	SDCKE	P9			
	SDCK	U7			
	CAS	U8			
	RAS	N10			
	SDWE	T8			
	NCS1/SDCS	M9		片选信号	
	SDA10	N6			
FLASH	AB1~AB19			地址总线	

	DB0_~DB15			数据总线	
	NRD/NOE/CFOE	N9			
	NWR0	P8			
	NCS0/BFCS	U6		片选	
SERIAL FLASH	PD6			片选	
	PA0				
	PA1				
	PA2				
	PA3				
CF	DB0~DB15			数据总线	
	AB0~AB10，AB22			地址总线	
	NWR0	P8			
	NRD/NOE/CFOE	N9			
	NWR1	T7			
	NWR3	R8			
	PC10				
	AB25				
	PC5				
	PC11				
	PC12				
	PB0				
	PC6				
Boot Mode	PA31			启动模式选择	
BUZZER	PB6				
LED	PB2				
SM	DB0~DB7				
	AB21，AB22				
	PC0，PC1，PC2，PC3				
	PB1				
	NCS3/SMCS	R7		片选	
SD	PA27				
	PA28				
	PA29				
	PB5				
IrDA	PA22				
	PA23				
USB Device	DDM				
	DDP				
	PD04			USB 检测	

	PD05			通讯线上拉控制	
USB HOST	HDMA				
	HDPA				
	HDMB				
	HDMA				
UART0	PA31			TX	
	PA30			RX	
UART1	PB26			RTS	
	PB20			TXD	
	PB19			DTR	
	PB23			DCD	
	PB25			DSR	
	PB21			RXD	
	PB24			CTS	
	PB18			RI	
ETHERNET	PA8， PA9，PA10				
	PA11 ， PA12 ， PA13				
	PA14，PA15，PA16				
	PC4				
AC97	PB10			BIT_CLK	
	PB11			SYNC	
	PB9			SDATA_IN	
	PB8			SDATA_OUT	
	PB27			XTAL_EN	
VGA Interface	AB1~AB20				
	DB0~DB15				
	NRD/NOE/CFOE	N9			
	NWR1	T7			
	NWR0	P8			
	PC6				
	AB21				
	NCS2				
DDA	PA24				
	PB17				
	PB16				
	PB15				
	PB14				
	PB13				

	PB12				
SPI Interface	PA0				
	PA1				
	PA2				
	PA3				
	PA4				
	PC13				
RS485	PA17				
	PA18				
	PC14				

2.4 模块电路介绍

EMBEST ATEB9200 评估板基于 AT91RM9200 芯片片上资源，整个评估板系统资源非常丰富，设计简洁。以下重点介绍一下评估板的启动模式电路，评估板的地址空间映射和 GPIO 外扩接口电路。

2.4.1 启动模式

AT91RM9200 处理器有两种启动模式：内部 ROM 启动和外部存储器启动。由跳线 SW801 状态来选择不同启动方式，当跳线连接 1-2 时选择从外部 NOR Flash 存储器启动，当连接跳线 2-3 时从内部 ROM 启动。具体电路如图 2-3 所示。

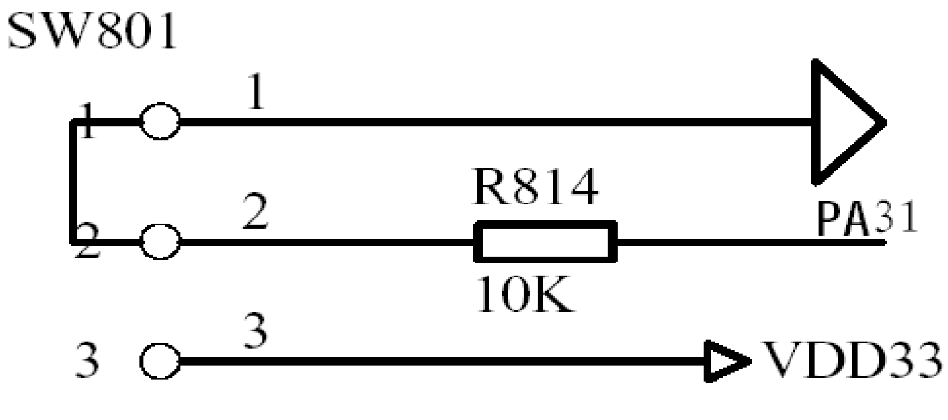

图 2-3

2.4.2 存储器空间映射

AT91RM9200 处理器的地址映射根据外部跳线 SW801 的跳线位置不同，其地址映射也不一样。在没有 REMAP 命令前地址 0 映射是根据 SW801 跳线状态决定的，当连接 SW801 跳线 1-2 时，从外部存储器 Nor Flash 启动，外部总线空间 0x10000000 内容映射到 0 地址。当 SW801 连接 2-3 时，处理器从内部 ROM 启动，内部 ROM 地址 0x100000 映射到 0 地址空间。执行 REMAP 命令后，不管 SW801 的跳线状态怎样，处理器都将内部 SRAM 地址空间映射到地址 0 空间。

当 SW801 的跳线连接 1-2 的时候，REMAP 前后地址空间映射对比如下图 2-4。

地址	REMAP 前	地址	REMAP 后
0x00000000 0x000FFFFF	外部 FLASH	0x00000000 0x000FFFFF	内部 SRAM
0x00100000 0x001FFFFF	内部存储区 ROM	0x00100000 0x001FFFFF	内部存储区 ROM
0x00200000 0x002FFFFF	内部 SRAM	0x00200000 0x002FFFFF	内部 SRAM
	… …		… …
0x10000000 0x101FFFFF	外部 FLASH	0x10000000 0x101FFFFF	外部 FLASH
0x20000000 0x21FFFFFF	外部 SDRAM	0x20000000 0x21FFFFFF	外部 SDRAM

图 2-4

当 SW801 的跳线连接 2-3 的时候，REMAP 前后地址空间映射对比如下图 2-5。

地址	REMAP 前	地址	REMAP 后
0x00000000 0x000FFFFF	内部存储区 ROM	0x00000000 0x000FFFFF	内部 SRAM
0x00100000 0x001FFFFF	内部存储区 ROM	0x00100000 0x001FFFFF	内部存储区 ROM
0x00200000 0x002FFFFF	内部 SRAM	0x00200000 0x002FFFFF	内部 SRAM
	… …		… …
0x10000000 0x101FFFFF	外部 FLASH	0x10000000 0x101FFFFF	外部 FLASH
0x20000000 0x21FFFFFF	外部 SDRAM	0x20000000 0x21FFFFFF	外部 SDRAM

图 2-5

2.4.3 总线接口扩展接口

为了方便客户在 Embest ATEB9200 开发板上外扩自己的硬件，开发板将总线和全部 IO 外扩出来，其具体接口定义如下图 2-6 和图 2-7。

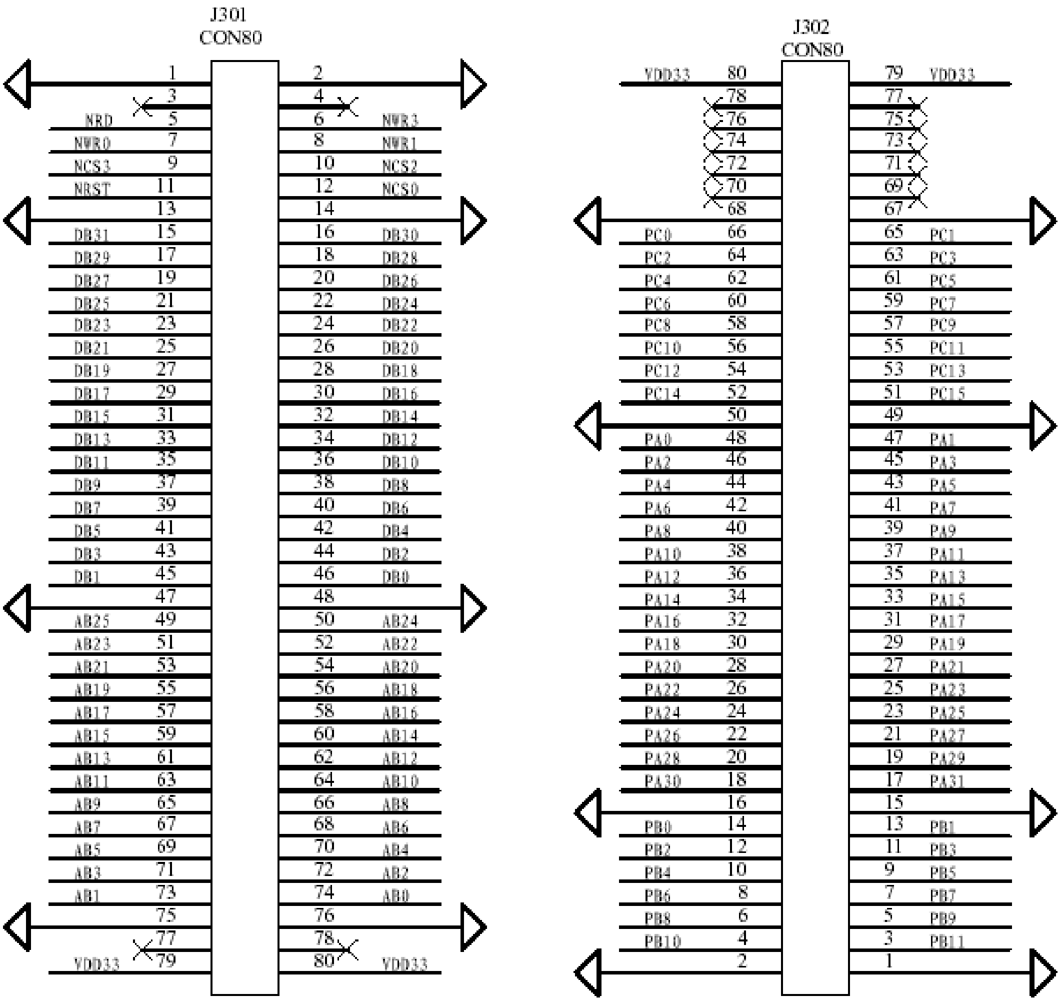

图 2-6

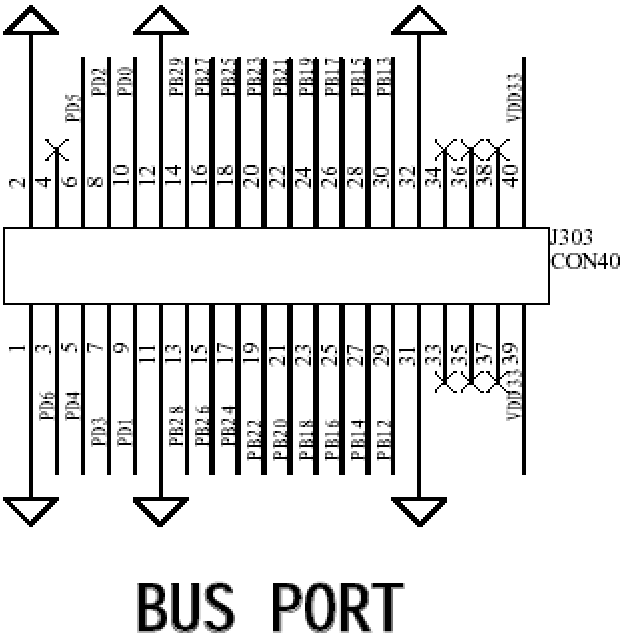
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
PD5
PD2
PD0
PB29
PB27
PB25
PB23
PB21
PB19
PB17
PB15
PB13
VDD33
J303
CON40
1
3
5
7
9
11
13
15
17
19
21
23
25
27
29
31
33
35
37
39
PD6
PD4
PD3
PD1
PB28
PB26
PB24
PB22
PB20
PB18
PB16
PB14
PB12
VDD33
BUS PORT

图 2-7

第三章 EMBEST ATEB9200 开发板安装使用

3.1 静电警告

板上器件多为静电敏感器件，请勿用手直接触摸板上任何金属部分，以便静电损伤电子器件，如果需要接触板上金属部分，请佩戴防静电手环或采取一定的防静电措施。

3.2 开发板装箱清单

- EMBEST ATEB9200 评估板 1 块
- 交叉串口线 1 条
- 交叉网线 1 条
- USB 线 1 条
- 5V 1A 电源 1 个
- 配套光盘 1 张

3.3 开发板上电默认跳线设置

跳线设置		功能描述
SW801	1-2 连接	外部 NOR Flash 存储器启动
SW201	1-2 连接	串行 DataFlash 使能
SW402	1-2 连接	JTAG 口使能

3.4 开发板上电

EMBEST ATEB9200 评估板使用外接稳压电源通过一个 2.1mm 的插座进行供电（5V DC），当使用外接稳压电源对评估板进行供电之前，请务必确认该稳压电源的输出极性为“内正外负”！具体见图 3-1 所示。

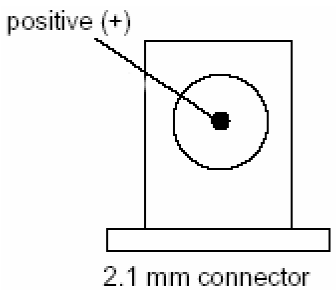

图 3-1

当接通电源以后，电源指示灯 D201 会被点亮，如果该指示灯不亮，请迅速关掉电源，并检查电源连接情况。

3.5 连接开发板

3.5.1 超级终端配置

使用配套的交叉串口线连接开发板的 UART0 和 PC 电脑串口，按照如下图（图 3-2）配置超级终端。

图 3-2

3.5.2 上电前检查

- 开发板使用的电源是否正确，应该为 5V DC 电源，电源内正外负。
- 开发板是否有明显的短路，受潮等异常现象。
- 将开发板的电源开关拨至 OFF 状态，连接开发板电源。

3.5.3 开发板上电

将开发板的电源开关拨至 ON 状态，注意电源指示灯 D201 是否同时点亮，同时观察到超级终端会打印如下信息，见图 3-3。

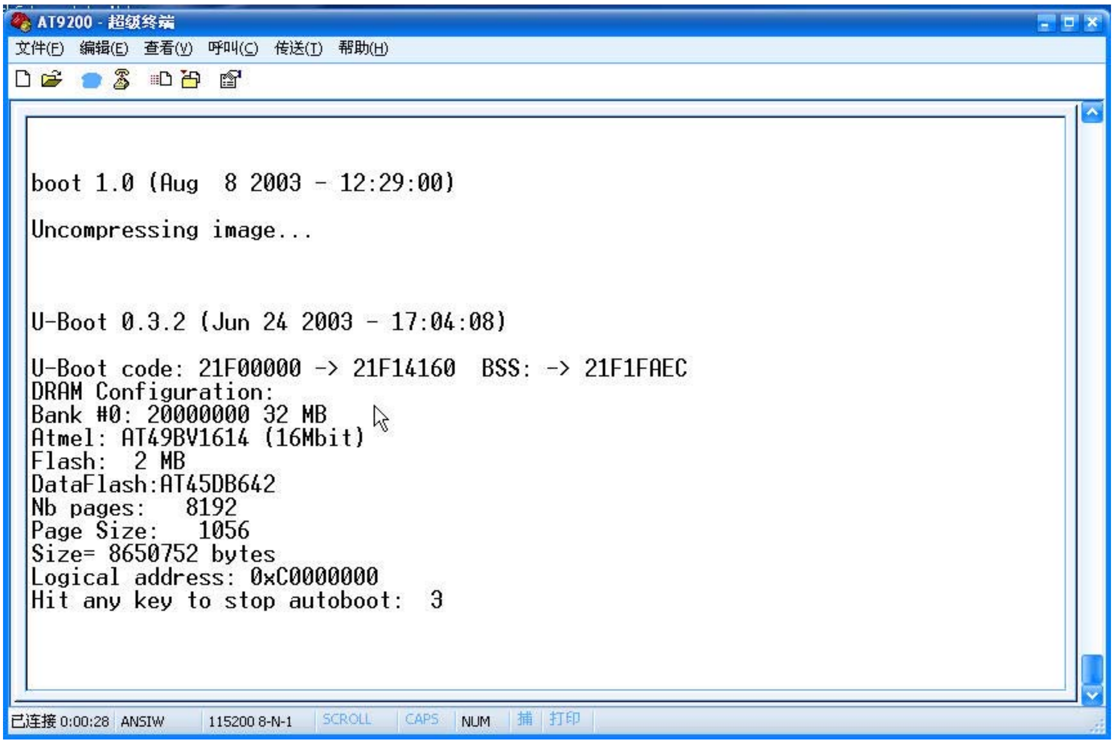

图 3-3

上图是启动 Uboot 的过程，当显示“Hit any key to stop autoboot:3”时，如果用户通过超级终端有键盘输入，则 Uboot 进入 Uboot 的命令提示符界面，能够进行 Uboot 的相关命令操作，具体见第五章。

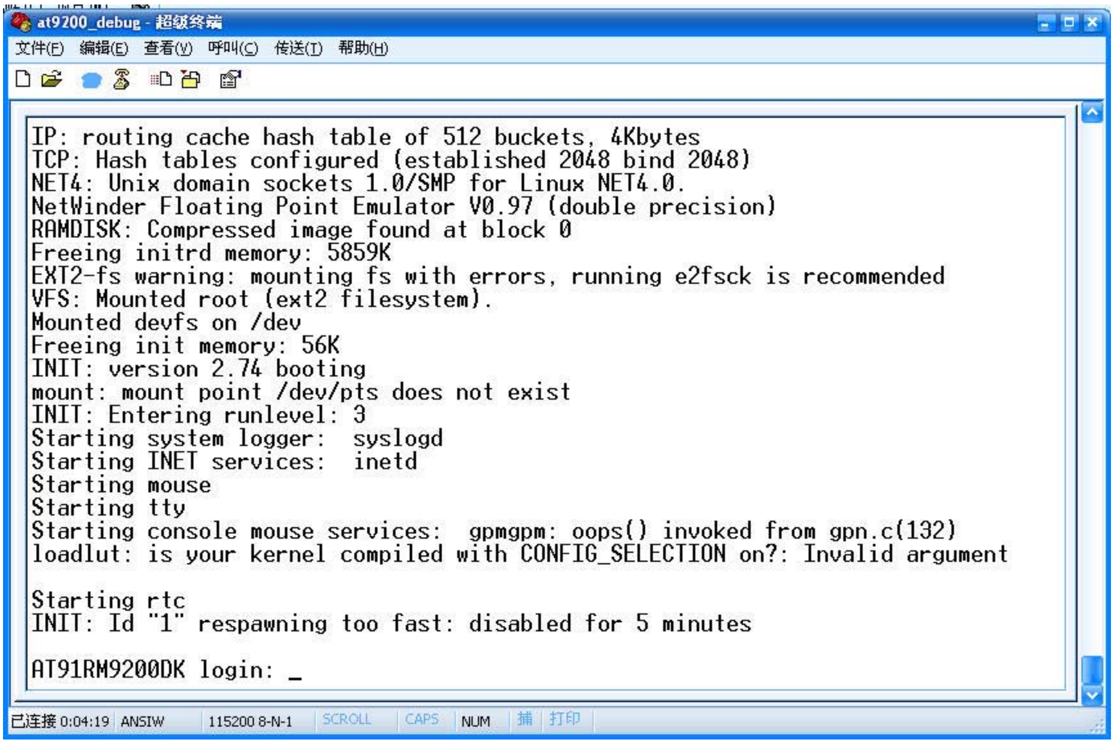

图 3-4

当显示“Hit any key to stop autoboot:3”时，如果 3 秒钟用户通过超级终端没有键盘输入，开发板会启动到 linux 下，如图 3-4 所示。接着用户可以 root 登陆到 linux 系统（登陆无需密码），可以进行 linux 下的相关操作。

3.6、光盘内容简单介绍

光盘结构如下

```
├── Products          Embest 产品介绍
├── Doc               EMBEST ATEB9200 评估板资料
├── Software          基于 EmbestIDE 示例程序
├── Tools             相关工具
├── ATEML 相关资料    ATEML 原版光盘相关资料
├── Uboot             Uboot 源码及编译 Uboot 核
└── Linux             linux 源码、编译 linux 核及文件系统
```

- Products 文件夹为 Embest 公司产品介绍，系统地介绍 Embest 的现有的评估板，ARM 仿真器开发套件，Flash 烧写软件，ARM 培训等资料。
- DOC 文件夹包含 Embest ATEB9200 开发板的用户手册，评估板相关 IC 资料，评估板原理图，评估板的 PCB 封装库等。
- Software 文件夹下为 Embest 公司 IDE 下开发的裸机调试程序。包含串口，USB，MMU，以太网，SPI 等基本模块功能示例程序。
- Tools 文件夹主要是一些开发 ATEB9200 评估板过程中需要使用的工具，比如：ARM 交叉编译工具，Embest EDU 软件和 tftp 工具等。
- ATEML 原版资料文件夹下包含 ATEML 公司原版英文光盘的资料。其中提供的 html 文本中较详细的介绍了 Uboot 和 linux 编译和使用过程。
- Uboot 文件夹下包含 Uboot 程序源代码和编译好的能够直接使用的 Uboot 内核代码。
- Linux 文件夹提供了经过我们修改的 linux 源代码和编译好的 linux 内核代码。

第四章 启动过程分析

整个开发板上电启动过程分四阶段，分别是：芯片 boot 选择，boot 代码引导阶段，Uboot 引导阶段和最终的 linux 引导阶段。

4.1 芯片启动模式选择

系统启动，首先根据 SW801 的跳线设置，当连接 1-2 的时候 0x10000000（分配的是 Nor Flash）地址重映射到 0x00000000 地址，而 Nor Flash 中烧写的是 Uboot 程序，因此 Uboot 会启动，具体启动过程见下节。当连接 2-3 时，首先启动芯片 ROM 的 bootloader 程序，bootloader 程序首先依次检查 SPI 接口的 Dataflash，TWI 的 E2PROM 和连接 EBI 的 8 位并行存储器是否存在有效的中断向量表，如果找到有效的中断向量表，就会将对应的向量表拷贝到 SRAM 中，然后跳到中断向量表执行。最后如果找不到合法的中断向量表，则运行一个 Uploader 程序，此时会初始化 DBG 口，在超级终端不停打印 CCCC，用户可以使用 xmodem 协议通过超级终端下载 Uboot 程序到 SRAM 中。

以下以外部存储器启动方式为例介绍开发板启动过程。

4.2 boot 引导过程

当跳线 SW801 连接至 1-2 时，芯片从外部存储器启动。

整个 2M 的 Nor Flash 的存储内容分配如下：boot Image 代码，Uboot 的环境变量和 Uboot 代码三部分组成，具体地址分配的地址空间如下，见图 4-1。

当系统从外部存储器启动时，首先运行位于 0x10000000 的 boot 代码。Boot 代码主要负责初始化系统的基本的软硬件环境件环境，将压缩 Uboot 代码解压到 SDRAM 中，然后跳转到 Uboot 代码的开始地址，运行 Uboot 代码。

4.3 Uboot 引导过程

Uboot 首先初始开发板，然后通过 SPI 接口从 DataFlash 中将 linux 拷贝到 SDRAM 的 0x21000000，将文件系统拷贝到 SDRAM 的 0x21100000 地址，然后跳转到 0x21000000 地址，Uboot 首先检查 linux 的文件头，进行 CRC 校验，如果正确，跳到 linux 内核执行。

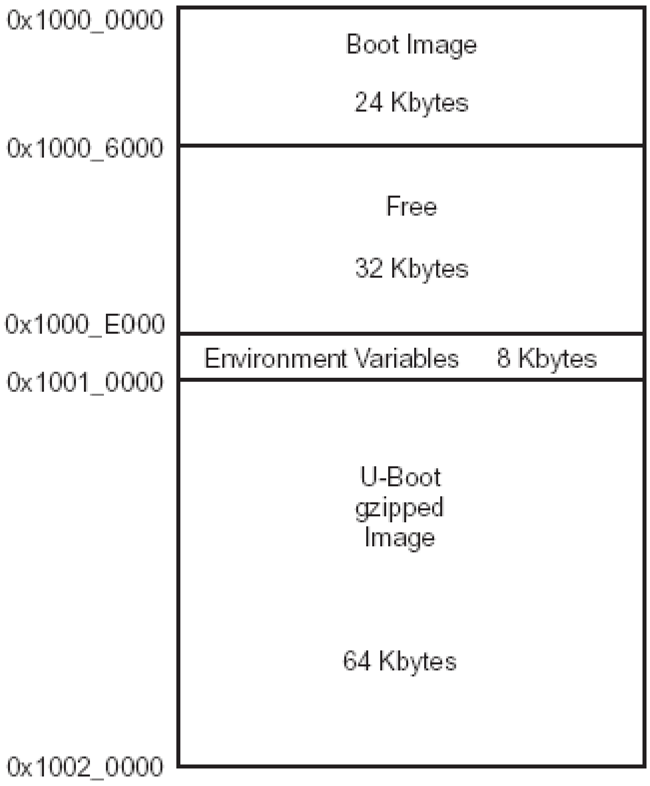

图 4-1

4.4 Linux 引导过程

当从 Uboot 跳转到 linux 后，首先建立堆栈并解压缩内核映象文件；接着进行数据区、终端描述表、段描述表、页表和寄存器的初始化，进行完一些必要的状态检查后，转入到 start_kernel()函数的继续初始化工作，如：linux 版本信息打印，设置异常事件、系统调用和调用门入口地址，初始化中断，初始化控制台和显示器，初始化 PCI，初始化网络等；初始化完内核后，开始用户态初始化工作，如内核加载文件系统，创建进程支持用户登陆，获取登陆信息和检验用户，最后启动 shell，允许用户交互地执行用户命令。

第五章 Uboot 安装与应用

5.1 Uboot 简单介绍

Uboot 是一个通用的免费开放源码的 boot 程序，支持很多的处理器，相当于 PC 机的 BIOS 但是又远远的强于普通的 BIOS，比如支持网络引导，引导各种内核，甚至一个简单的 shell，等等，是基于 GPL 的开源自由软件。

下载地址：http://sourceforge.net/projects/u-boot，用户可以去网站下载最新的 Uboot 源代码。

我们产品光盘\uboot\UBoot source 文件夹也提供了 Uboot 的源码和编译好的 Uboot 烧写文件，用户可以直接使用。

5.2 Uboot 编译

5.2.1 环境要求

1、 要求在 linux 操作系统环境下编译 Uboot。

2、 要求安装交叉编译工具。（如果用户有安装交叉编译工具，具体安装过程请参考第六章的 6.2.1 节，关于编译工具安装介绍）。

5.2.2 编译步骤

以下以我们产品光盘提供的 Uboot 源代码为例讲述 Uboot 的编译步骤。

1、 以 root 身份登陆 linux，在/usr/src 目录下新建 Uboot 目录。

2、 将光盘里的提供的源代码解压到/usr/src/uboot。具体命令：

tar –zxfv /mnt/cdrom/uboot/uboot source/AT91RM9200-Boot.tgz。

3、 解压后在/usr/src/uboot 文件夹下生成 AT91RM9200-Boot 文件夹。文件夹由 AT91RM9200-Boot，AT91RM9200-Loader 和 AT91RM9200-U-Boot 组成。直接在 AT91RM9200-Boot，和 AT91RM9200-Loader 路径下输入 make 命令，就能生成对应的 boot.bin 和 loader.bin 二进制文件。

4、 编译 Uboot。在 AT91RM9200-U-Boot 路径下，输入命令：

- make at91rm9200dk_config ；配置成 at91rm9200 开发板环境
- make all ；编译 Uboot，生成 u-boot.bin 文件。
- gzip –c u-boot.bin > u-boot.gz ； 压缩 Uboot 二进制文件

以上就是 Uboot 的编译过程，通过以上步骤，生成 boot.bin，loader.bin，u-boot.bin 和 u-boot.gz 四个文件，接下来就可以烧写 Uboot 了。

5.3 烧写 Uboot 介绍

在评估板上首次烧写 Uboot 程序大致分为两个步骤：通过 Xmodem 协议将 Uboot 程序下载到 SDRAM 中，然后通过 Uboot 命令将被烧写的 Uboot 代码烧写到 Nor Flash 或者 DataFlash 中。

5.3.1 启动 Uboot

打开超级终端(参数设置为 115200，8，N，1)，短接 SW801（2-3）跳线（启动芯片内部的 Uploader 程序），并给评估板上电。超级终端显示如图 5-1，不停接收到 cccc 字符(如图 5-1)，就表明目标板已准备好，允许通过超级终端将 Uboot 程序下载到 SDRAM 中。

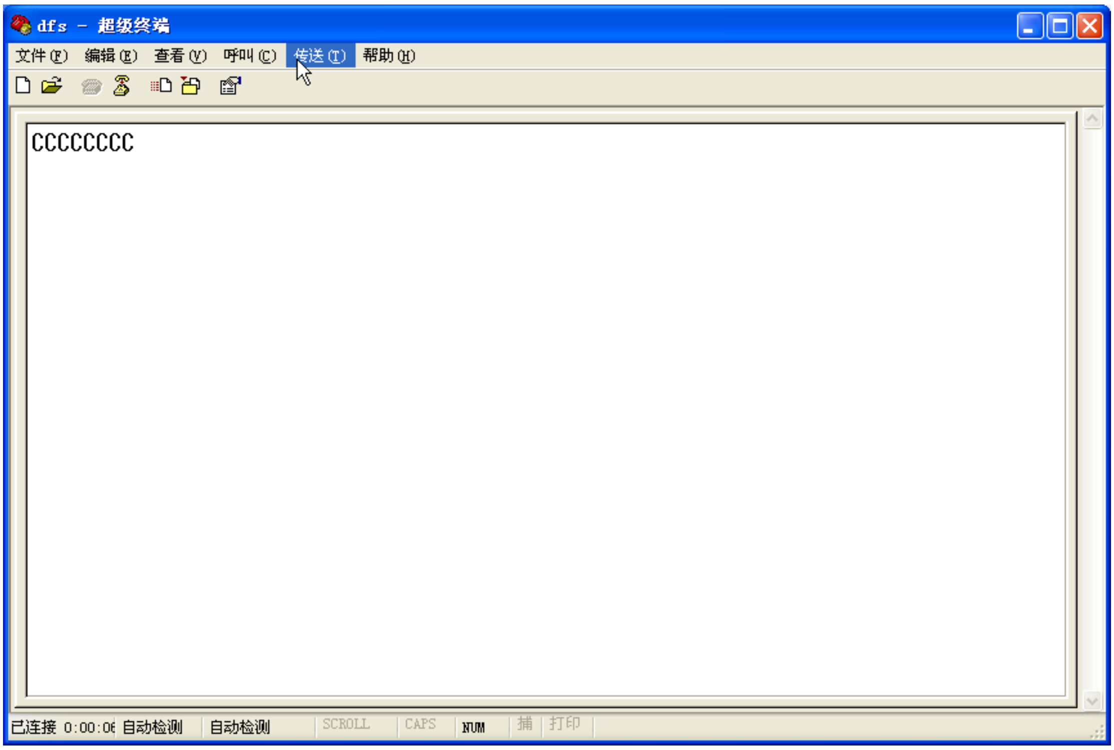

图 5-1

下载 Uboot 到 SDRAM 中运行的需要下载两个文件，分别是 loader.bin 和 u-boot.bin。此时点击传送－＞发送文件，选择 Uboot 文件夹下的 loader.bin 文件以 XMODEM 方式发送，如图 5-2 所示。

图 5-2

文件发送完毕后，程序会执行 Loader1.0 程序。此时会提示用户继续发送 U-boot 程序文件，同时继续不停的打印 CCC。如图 5-3。

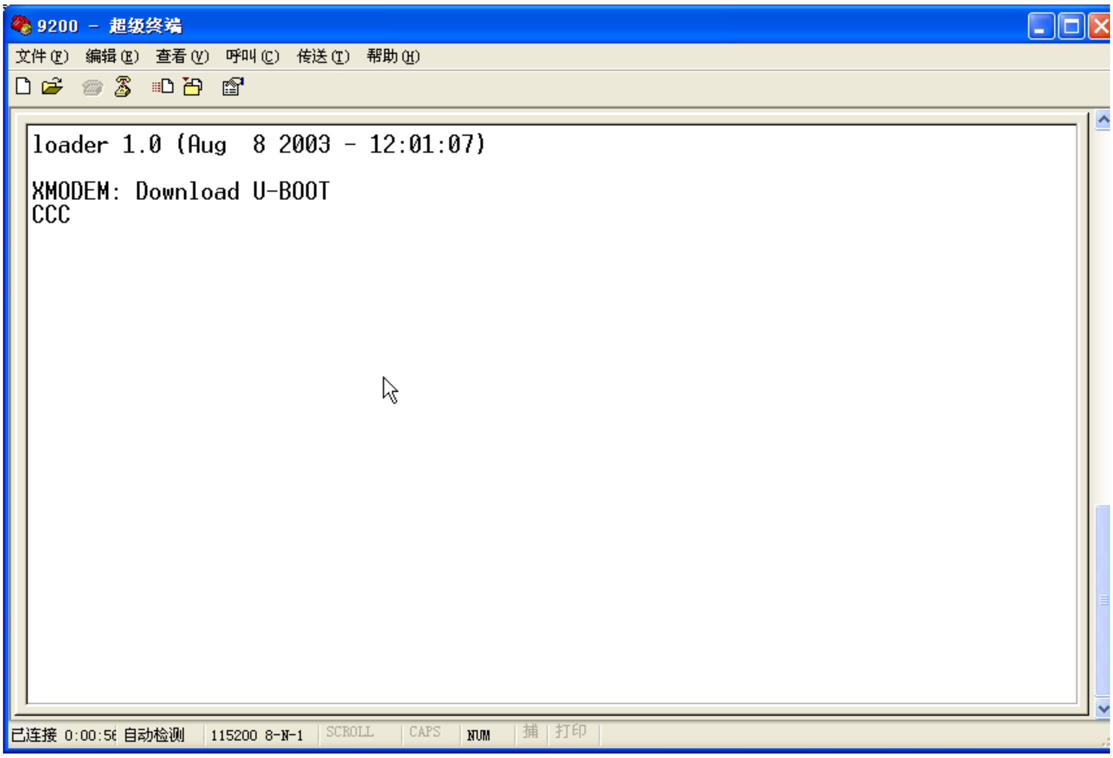

图 5-3

按照发送 loader.bin 的方式同样地将 u-boot.bin 下载到 SDRAM 中，下载完后 Uboot 会自动运行，其界面如图 5-4。

5.3.2 烧写 Uboot

前面步骤并没有把程序装入到 Flash 中，而只是下载到内存，然后在内存中运行。下面我们使用 Uboot 中的命令将程序烧录到 Flash 中。烧写到 Flash 的文件有两个，分别是 boot.bin 和 u-boot.gz。

1、 烧写 boot.bin

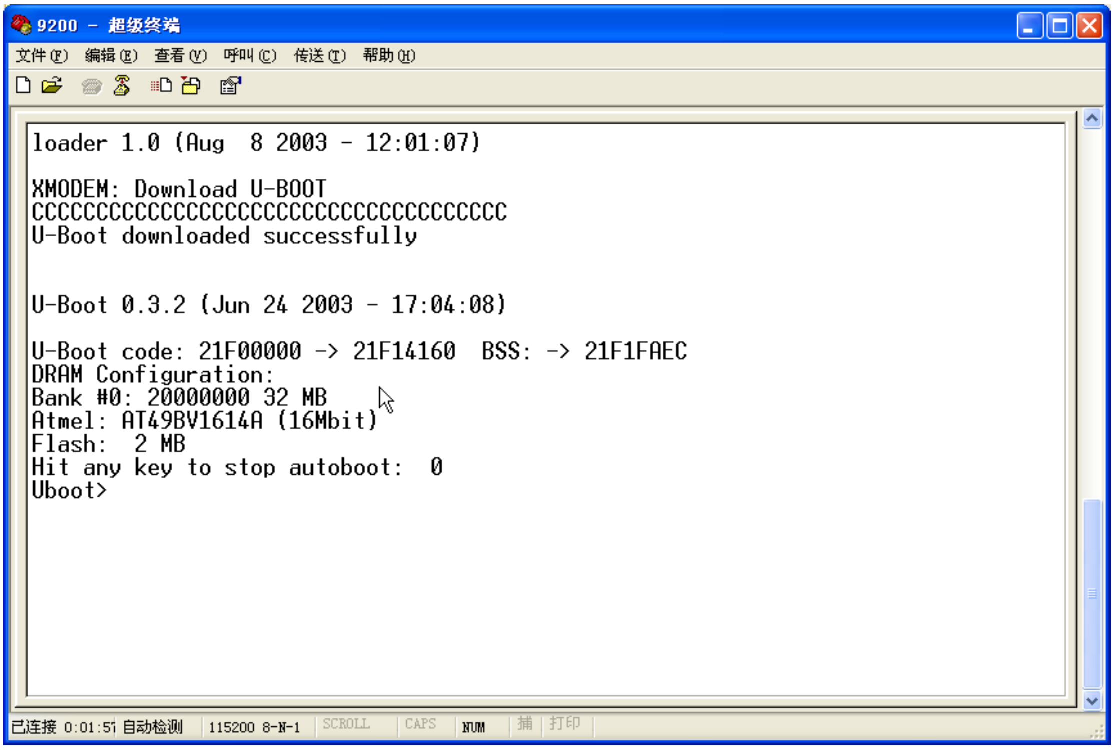

图 5-4

boot.bin 的在 flash 中的烧写地址空间为 10000000 到 10005ffff，具体的烧写步骤如下。

- loadb 20000000　　；　将文件 boot.bin 装载到 20000000 地址（如图 5-5 所示）

```
## Ready for binary (kermit) download to 0x20000000 at 115200 bps...
_
```

图 5-5

- 通过串口的 Kermit 协议发送 boot.bin 文件。
- protect off 10000000 10005fff　；　去写保护
- erase 10000000 10005fff　；　擦除数据
- cp.b 20000000 10000000 5fff　；　命令将 20000000 中的数据拷贝 0x5fff 字节到 10000000 地址中
- protect on 10000000 10005fff　；　给 Flash 加保护

完成过程如图 5-6 所示。

2、烧写 u-boot.gz

烧写 u-boot.gz 的地址空间为 10010000 到 1001ffff，具体烧写步骤如下：

- loadb 20000000　　；　传送文件 u-boot.gz 到 SDRAM 的 0x20000000 地址。

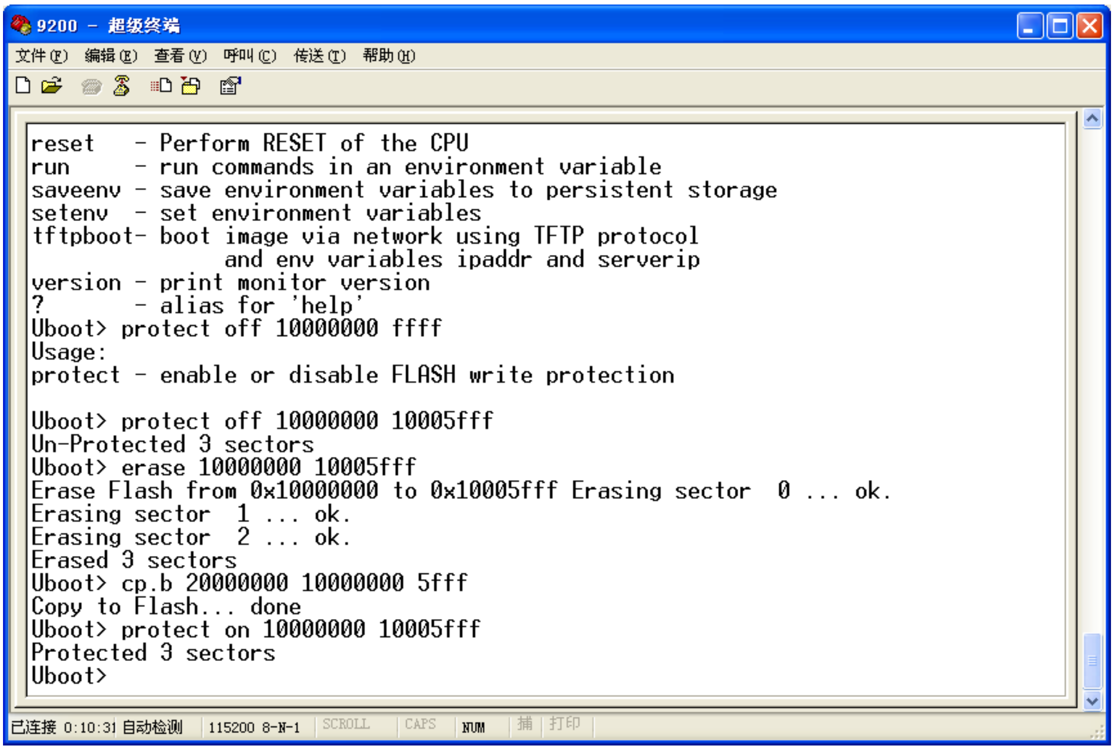

图 5-6

- 通过串口的 Kermit 协议发送 u-boot.gz 文件
- protect off 10010000 1001ffff ； 去 Flash 写保护
- erase 10010000 1001ffff ； 擦除 Flash
- cp.b 20000000 10010000 ffff ； 拷贝数据到 Flash

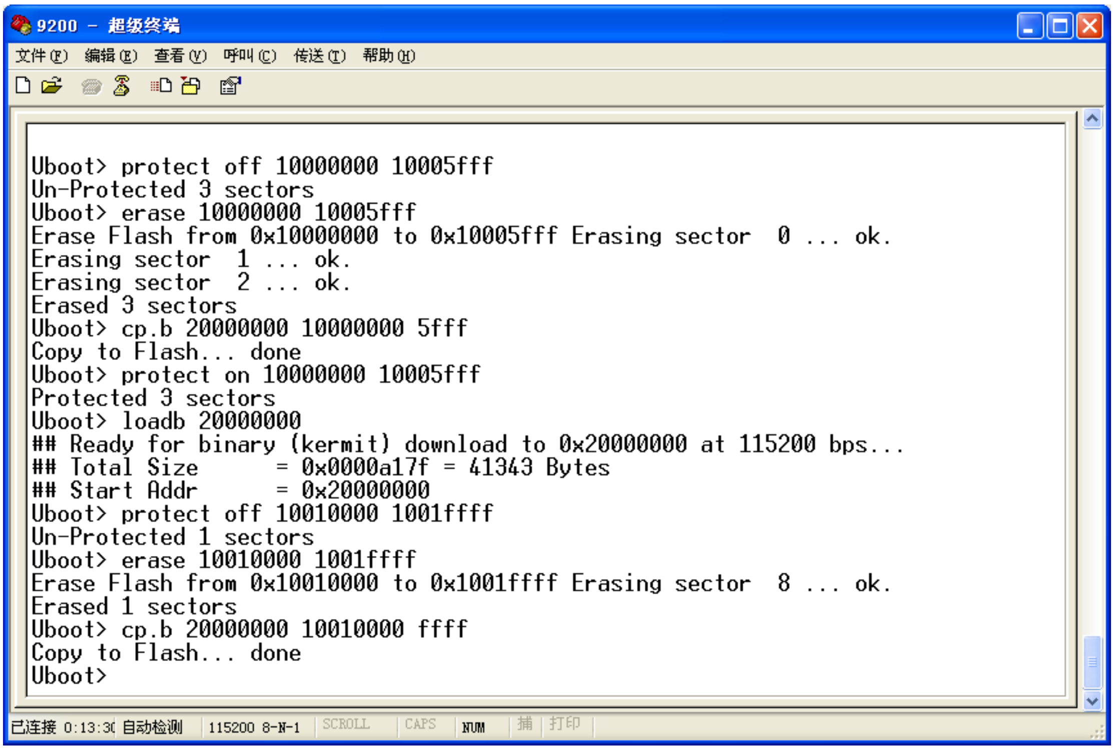

图 5-7

- protect on 10010000 1001ffff ； 给 Flash 加保护

完成过程如图 5-7。

整个烧写过程就完成了，关闭电源，连接跳线 SW801（1-2）跳线，选择从外部存储器启动，然后给开发板上电，系统就会启动 Uboot 程序，启动界面如图 5-8。

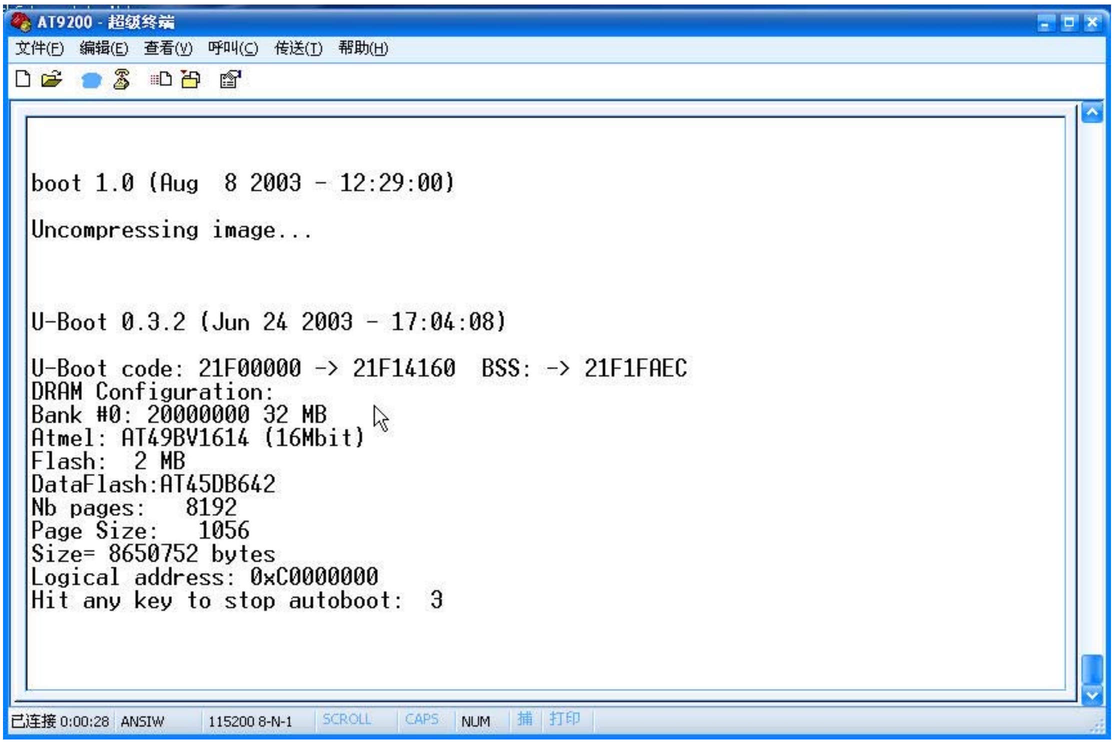

图 5-8

5.4 Uboot 常用命令介绍

- loadb

功能：通过串口 kermit 协议下载文件到 SDRAM。
语法：loadb [地址]

- cp.b

功能：从 ram 拷贝文件到 flash。
语法：cp.b [ram 地址] [flash 地址] [十六进制文件大小]

- erase

功能：擦除 flash
语法：erase [起始地址] [结束地址]

- setenv

功能：设置环境变量
语法：setenv [变量名称] [命令]

- go

功能：跳转到一个地址处执行程序。
语法：go [地址]

➢ bootm

功能：启动操作系统
语法：bootm [ram 地址]

➢ tftp

功能：tftp 下传文件
语法：tftp [RAM 地址] 文件名

其他命令用户可以通过 help 命令在线查看。

5.5 Uboot 下的应用程序开发

Uboot 把操作系统和硬件平台衔接在一起，功能是如此的强大，为我们配置好了系统的时钟，外部存储器接口，串口，网卡等，使得我们不需要做非常复杂的对应硬件寄存器的配置，就可以使用这些资源。在没有仿真器情况下，我们就可以利用 Uboot 提供给我们的资源，在 Uboot 基础上进行应用程序开发。开发应用程序步骤：

5.5.1 应用程序开发

客户可以在 EmbestIDE 下或者 ADS 下编辑应用程序，在应用程序里通过一些串口打印调试信息，然后编译连接生成应用程序的影像文件。

5.5.2 下载应用程序

Uboot 下将程序下载到 SDRAM 中有两种方式：

1、 loadb 命令。通过串口的 kermit 协议下载到 SDRAM。

2、 tftp 命令。采用 TFTP 方式通过以太网下载到 SDRAM。

5.5.3 应用程序调试

将应用程序下载到 SDRAM 中后，使用 go 命令，直接跳转到应用程序的入口地址，执行指令，通过察看超级终端的调试打印信息调试程序。如果出现问题重复 5.5.1 节，直到成功。

第六章 Linux 安装使用介绍

6.1 linux 介绍

Linux 操作系统诞生于 1991 年的 10 月 5 日，是目前使用比较广泛的操作系统，Linux 最初设计为桌面系统，现在广泛应用于服务器领域。它支持多种体系结构，支持大量外设，网络功能完善，与 UNIX 系统兼容，开放源代码并有丰富的软件资源，内核稳定而高效，大小及功能均可定制。正是由于 Linux 自身的一些优良特性，能在很大程度上满足嵌入式操作系统的特殊要求，催生了一些嵌入式 Linux 系统，其中就包括 uClinux。

本产品光盘中也提供了从网上下载并且经过修改的 linux 源程序，文件路径：光盘\linux\linux-2.4.26.tar.gz，同样在光盘下的 linux 目录下也有编译好的 linux 内核和文件系统，用户可以直接使用这些内核。

6.2 linux 编译

如果客户直接使用我们编译好的 linux 内核和文件系统可以跳过这节，直接看 6.3 节内容。

注意：编译工具的安装，Uboot 编译和 linux 编译必须在 PC 机 linux 操作系统平台下。

6.2.1 编译工具安装

产品光盘提供的交叉编译工具路径为：光盘\tools 文件夹下 cross-2.95.3.tar.bz2 文件。

安装步骤：

以下步骤在 PC 电脑的 linux 环境下进行。

1、 以 root 用户名登陆，在 linux 系统的/usr/local 文件夹下新建 arm 目录。

2、 在 arm 路径下通过 tar 命令解压交叉编译工具到新建文件夹 arm 目录下。使用命令为：tar jxfv /mnt/cdrom/tools/cross-2.95.3.tar.bz2。

3、 增加/usr/local/arm/2.95.3/bin 路径到系统环境变量 PATH 下。具体实现方法：使用 vi 编辑器打开/root 目录下的.brash_profile 文件，在此文件空白行增加：PATH=$PATH; /usr/local/arm/2.95.3/bin。

6.2.2 linux 编译具体步骤

现在以光盘中提供的 linux 源代码为例简单说明一下 linux 的编译步骤：

注意由于我们光盘中提供的源代码是经过修改过的，因此和 ATEML 公司原版光盘带的帮助文件中叙说的编译步骤有些差异，请用户在编译这两者时候注意区分步骤。

以下步骤在 PC 电脑的 linux 环境下进行。

1、 以 root 用户名登陆，在/usr/src 目录下新建 arm 目录。

2、 解压 linux 源代码到新建目录中。具体命令：tar zxfv /mnt/cdrom/linux/linux-2.4.26.tar.gz。

3、 进入到解压后的目录 linux-2.4.26。运行一下命令编译 linux：

- make menuconfig
- make clean
- make dep
- make uImage

4、 运行完以上命令后，在 linux-2.4.26 目录下生成一个 vmlinux 文件，接下来的工作还需要对生成的内核做一些压缩等处理。具体步骤如下：

- arm-linux-objcopy –O binary -S vmlinux linux.bin
- gzip -v9 linux.bin
- mkimage -A arm -O linux -C gzip -a 0x20008000 -e 0x20008000 -d linux.bin.gz uImage

注意：mkimage 是 Uboot 提供的工具，如果安装我们提供的 Uboot 源程序，则在 AT91RM9200-U-Boot\ tools 文件夹下。

6.3 采用 TFTP 方式启动 Linux

采用 tftp 方式启动 linux，原理是在 PC 机运行 tftp 程序，建立 tftp 服务器，然后开发板端通过 Uboot 的 tftp 命令将内核和文件系统下载到 SDRAM 中，然后跳转到内核的开始地址运行内核。步骤如下：

6.3.1 建立 TFTP 服务器

建立服务器步骤：

1、 在 windows 操作系统平台上，建立一个 tftp 目录，然后拷贝光盘\tools\TFTPSRV.EXE 到 tftp 目录下。

2、 拷贝\光盘\linux 文件夹下的 uImage.bin 和 ramdisk.bin 到 tftp 目录下。

3、 在 PC 机上运行 TFTPSRV.EXE 文件。如下图显示”TFTP Server successful started”表示 tftp 运行成功，如图 6-1 所示。

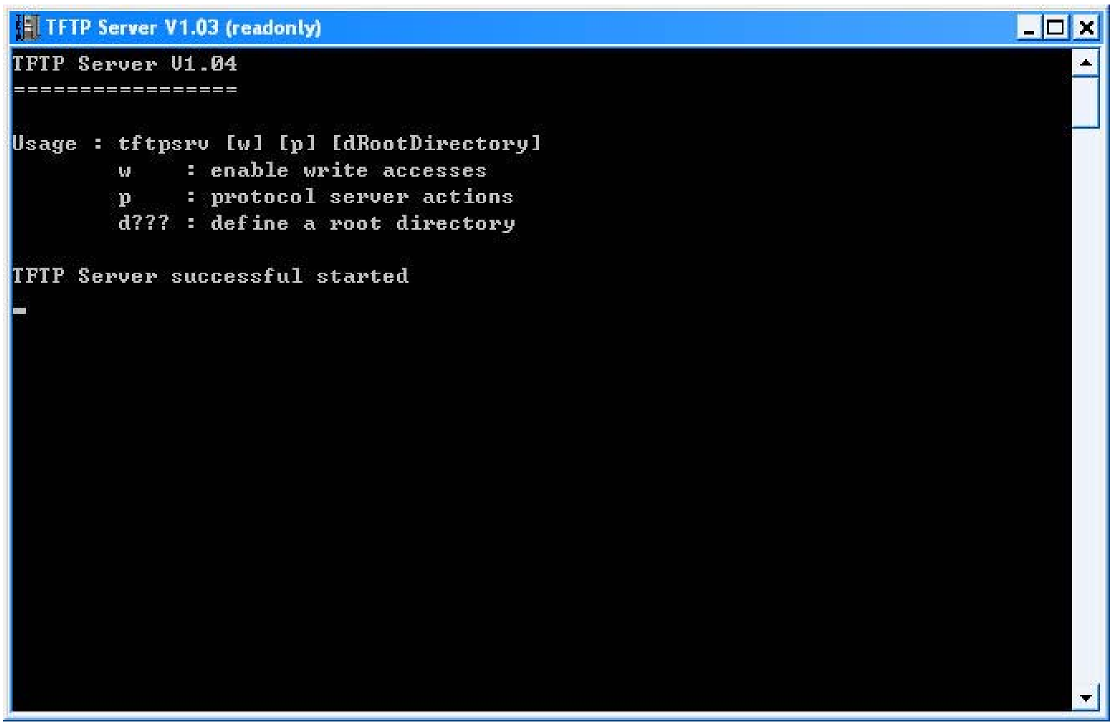

图 6-1

6.3.2 配置 Uboot

开发板启动 Uboot，按照如表配置 Uboot。配置的内容包括：串口的波特率，开发板网络的物理地址和 ip 地址，tftp 服务器 ip 地址，linux 启动参数配置。

baudrate	波特率　一般设置成 115200
ipaddr	评估板 IP。与服务器地址设置同一网段
ethaddr	评估板物理地址。（随意）
serverip	服务器 IP
bootargs	Linux 启动配制。一般配成 root=/dev/ram rw initrd=0x21100000,6000000 ramdisk_size=15360 console=ttyS0,115200 men=32M

我们可以通过 setenv 设定，printenv 查看。Saveenv 保存。

具体命令如下：

- setenv ipaddr 192.192.192.202
- setenv ethaddr 11:22:33:44:55:66
- setenv serverip 192.192.192.41
- setenv bootargs root=/dev/ram rw initrd=0x21100000,6000000

ramdisk_size=15360 console=ttyS0,115200,mem=32M

saveenv

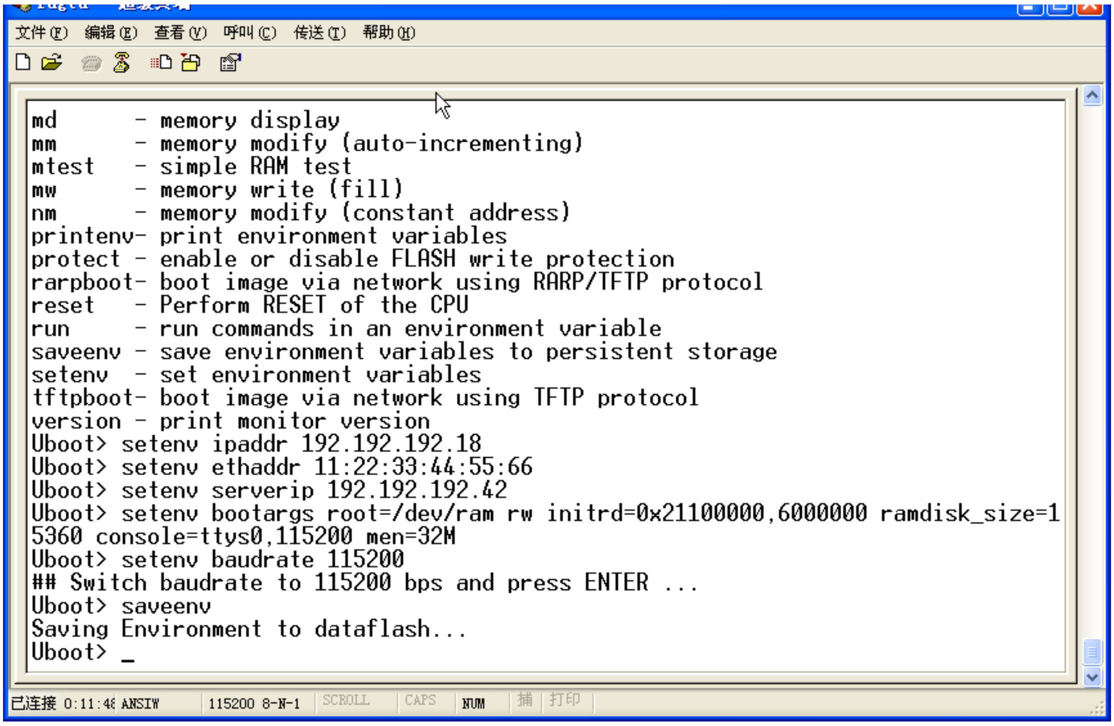

图 6-2 Uboot 配置过程图

3、下载内核

用 tftp 21000000 uimage 命令将 uimage 文件下载到 21000000 内存地址处。如下图 6-3。

4、下载文件系统

uimage 下载完毕后就使用命令 tftp 21100000 ramdisk 命令将 ramdisk 文件下载到 21100000 内存地址处。如下图 6-4。

5、启动 linux

到此在 SDRAM 中在 21000000 地址处有了 linux 内核，在 21100000 地址处有了 linux 文件系统，然后用 bootm 21000000 命令从 21000000 地址处的 linux 内核启动 linux。如图图 6-5。

```
Atmel: AT49BV1614A (16Mbit)
Flash:  2 MB
DataFlash:AT45DB642
Nb pages:   8192
Page Size:   1056
Size= 8650752 bytes
Logical address: 0xC0000000
Area 0: C0000000 to C0007FFF (RO)
Area 1: C0008000 to C001FFFF (RO)
Area 2: C0020000 to C0027FFF
Area 3: C0028000 to C083FFFF
In:    serial
Out:   serial
Err:   serial
Hit any key to stop autoboot:  0
Uboot> tftp 21000000 uImage
TFTP from server 192.192.192.42; our IP address is 192.192.192.18
Filename 'uImage'.
Load address: 0x21000000
Loading: #################################################################
         ################################################################
done
Bytes transferred = 655837 (a01dd hex)
Uboot>
```

图 6-3

```
Uboot> tftp 21100000 ramdisk
TFTP from server 192.192.192.42; our IP address is 192.192.192.18
Filename 'ramdisk'.
Load address: 0x21100000
Loading: #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         #################################################################
         ########################################################
done
Bytes transferred = 5607910 (5591e6 hex)
Uboot>
```

图 6-4

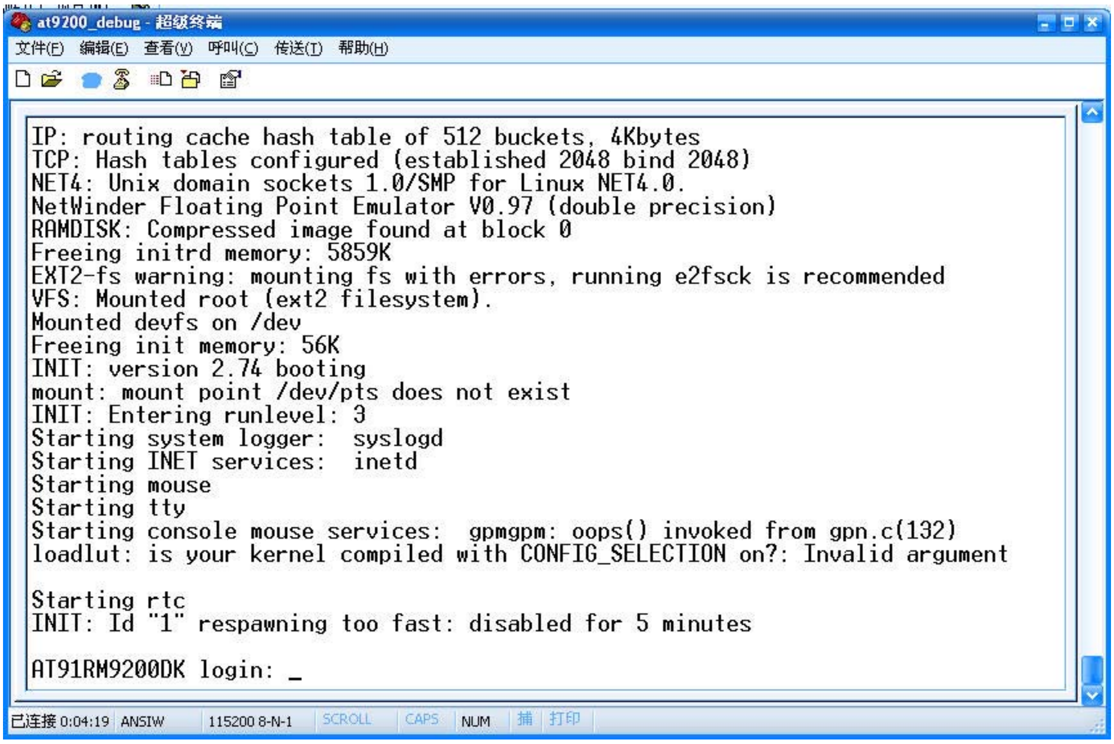

图 6-5

到此 linux 启动起来了，使用 root 用户登陆，然后客户就可以进行 linux 操作了，比如一些 linux 的基本操作，如图 6-6 所示。

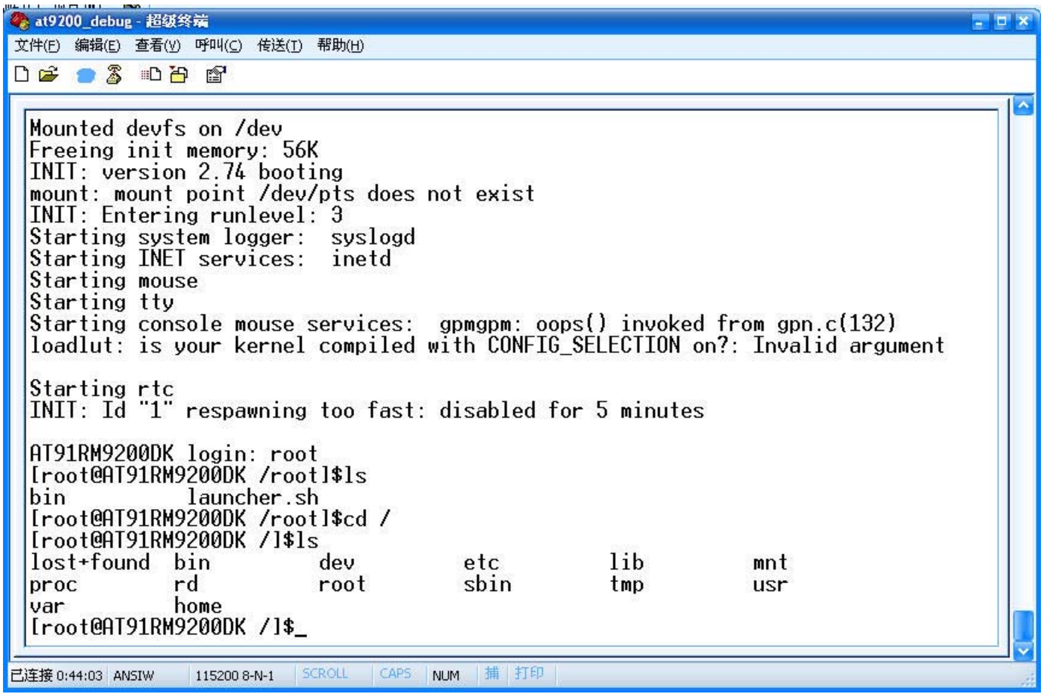

图 6-6

6.4 采用 DataFlash 启动 Linux

从 tftp 方式启动 linux 时，linux 内核和文件系统储存在 PC 机上，同样我们也可以将内核和文件系统烧写到 Flash 中去，那样就能够直接通过下载 Flash 里的内核和文件系统到 SDRAM 中运行了。内核编译完后大概 600 KB 左右，但是文件系统比较大，动辄几 MB，因此一般要求 Flash 比较大，一般将内核和文件系统存储在开发板系统的（8M）DataFlash 中，在准备将内核和文件系统烧写到 Flash 去前，先将**跳线 SW201 短接**。如上节介绍的那样，采用 TFTP 启动方式，将内核和文件系统下载到 SDRAM 后，在 Uboot 中通过 cp.b 命令将代码拷贝到 Flash 中。如图 6-7。

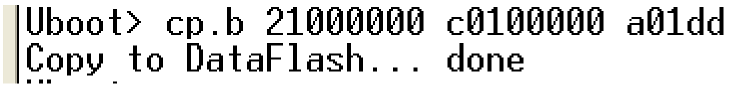

图 6-7

采用的命令如下：

- cp.b 21000000 c0100000 a01dd ； 拷贝内核到 DataFlash
- cp.b 21100000 c0200000 5591e6 ； 拷贝文件系统到 DataFlash

将内核和文件系统烧写到 Flash 后，启动 Uboot 后就可以将 Flash 中的两个数据区内容复制到 SDRAM 中，最后再用 bootm 21000000 方式启动 linux。命令如下：

- cp.b c0100000 21000000 a01dd
- cp.b c0200000 21100000 5591e6
- bootm 21000000

由于 Uboot 中的自动加载命令功能，有点像 DOS 中的自动加载批处理文件一样。DOS 中会在启动时自动加载 autoexec.bat 文件，而 Uboot 系统也是和此相似，会在 Uboot 启动后自动执行 bootcmd 环境变量里的命令。因此我们将上面的三条指令配置到 bootcmd 环境变量中，操作如下:

- setenv loadboot cp.b c0100000 21000000 a01dd
- setenv loadram cp.b c0200000 21100000 5591e6
- setenv bootcmd run loadboot\;run loadram\;bootm 21000000
- saveenv

给开发板重新上电，系统启动 Uboot 后，在默认的时间（3s）内没有命令输入，Uboot 自动直接执行从 Flash 拷贝内核代码到 SDRAM 中 21000000 位置，拷贝文件系统代码到 SDRAM 的 21100000 位置，然后执行 bootm 21000000 命令，开始跳转到内核位置，自动启动 linux。

第七章 裸机程序开发

在产品光盘提供了基于 EmbestIDE 集成环境开发的裸机程序，用户在 EmbestIDE 环境下编辑应用程序，编译连接生成应用程序影像，通过使用 Embest 公司的 ARM 仿真器，连接开发板和宿主机，将程序通过 EmbestIDE 直接下载到 SDRAM 中，然后运行，直接在 EmbestIDE 下察看运行结果。

7.1 使用方法

用户在使用示例程序前，首先需要在 EmbestIDE 的安装目录下的 examples 目录下新建一个 ATEB9200 项目文件夹，然后只需将光盘 Software 内的文件夹拷贝到此新建的文件夹即可，用户可以直接使用 EMBEST IDE for ARM 打开相应的工作区文件（*.ews），默认使用的编译器为 GNU Tools For Arm，稍作仿真器的参数设置即可。

注意事项：连接仿真器前确认，是否连接了 SW402 的 JTAG_EN（1-2）和 SW801 跳线 1-2。

7.2 调试过程示例

下面以 basic_test 为例，简单介绍一下调试过程:

首先在 EmbestIDE 安装目录的 examples 新建 ATEB9200 目录，然后从光盘拷贝\softwre 文件夹下的文件到 EmbestIDE\examples\ ATEB9200 目录下，然后打开 EmbestIDE 调试工具，点击 File-Open Workspace 菜单，打开 Open Workspace 对话框，选中 EmbestIDE\examples\ATEB9200\basic_test 文件夹并打开 basic.ews，如图 7-1。

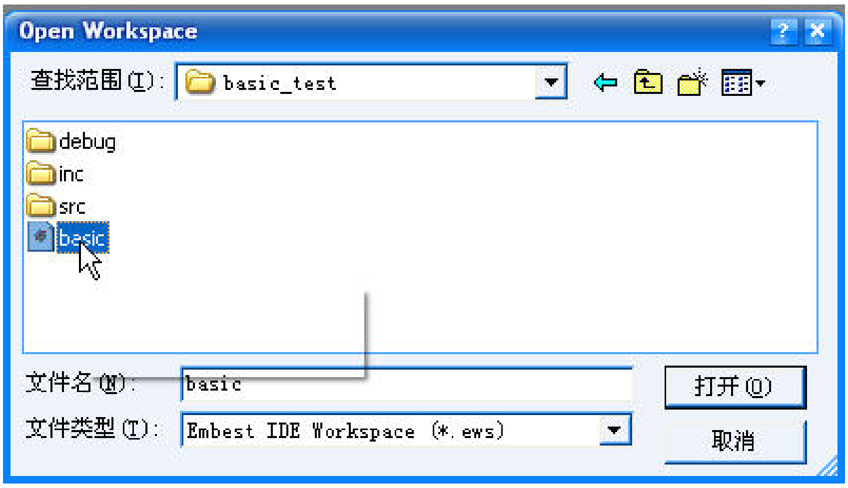

图 7-1

打开工程后下如图 7-2 所示。

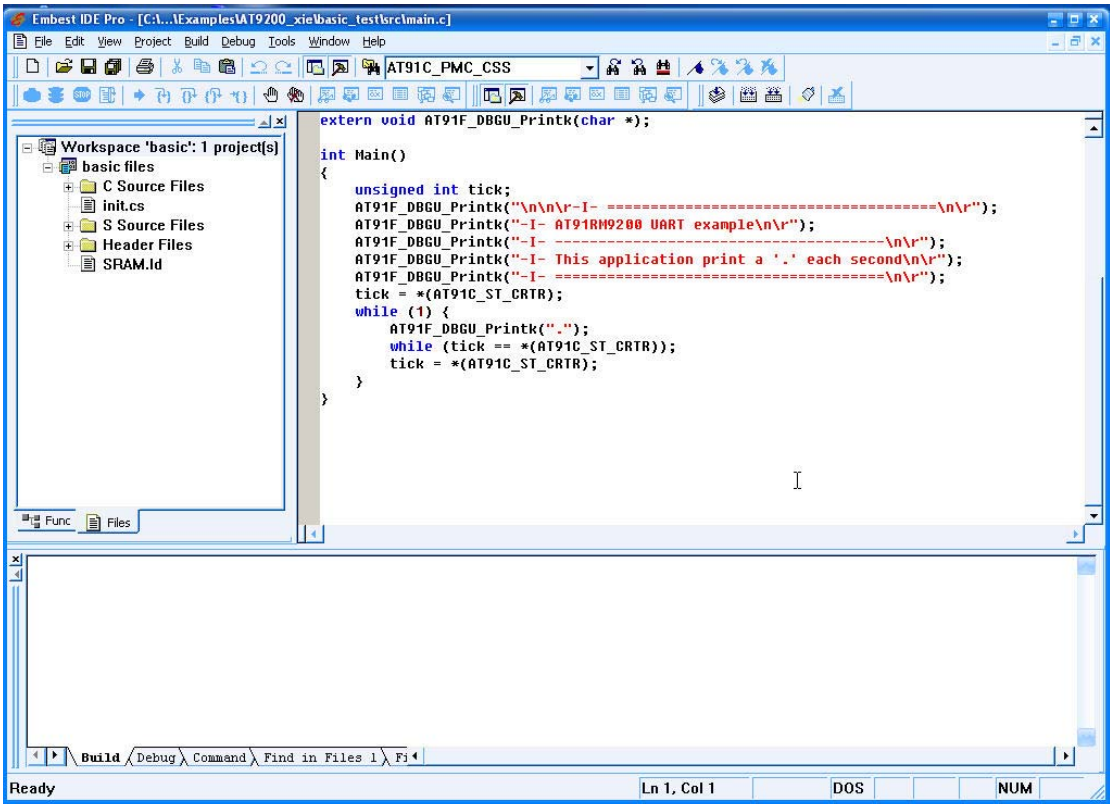

图 7-2

工程编译、链接，如图 7-3 所示。

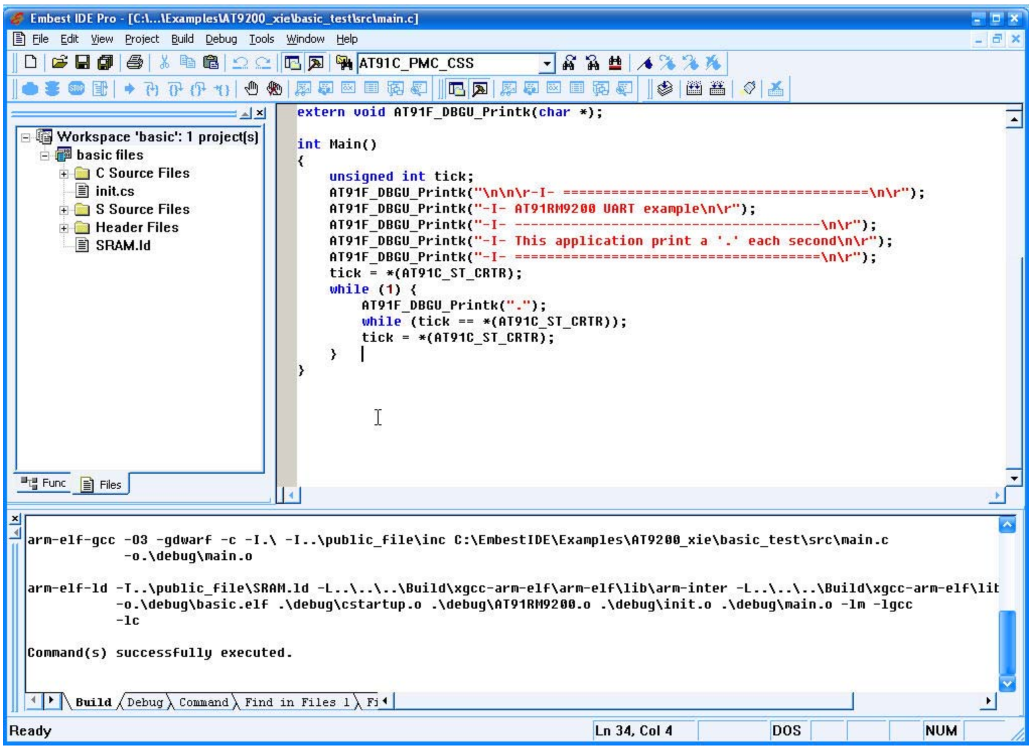

图 7-3

看到命令执行成功的提示信息，就可以开始在板上调试了。首先，连接仿真器和 EMBEST ATEB9200 评估板(注意如果使用 Embest Emulator 或者 Embest PowerICE，仿真器上的 14/20 开关，应拨到 20 一端)，然后连接仿真器和 PC 机。确认评估板电源正常后，就可以给评估板上电了。

在 SDRAM 中调试时需要一个命令脚本文件，在连接开发板时，EmbestIDE 根据命令脚本文件配置评估板的 ARM 处理器。如下图 7-4。

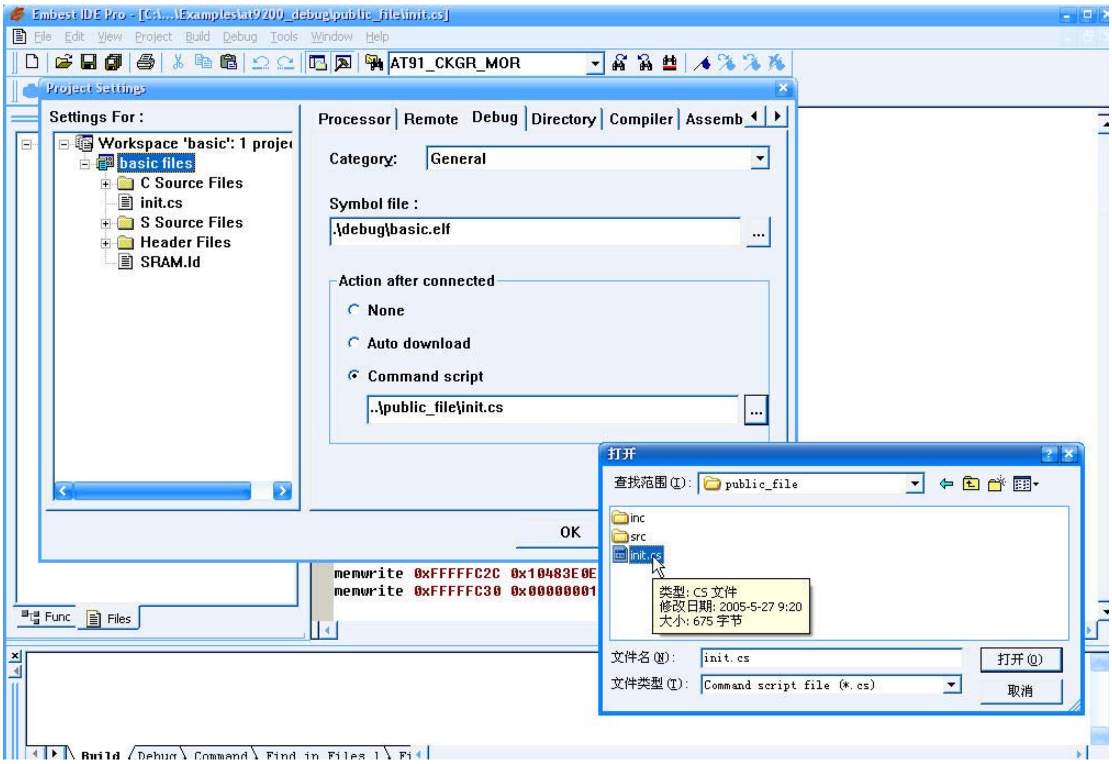

图 7-4

然后，点击 Project-settings 菜单或者通过热键 Alt+F7，打开 Project Settings 对话框。该对话框是开发、调试过程中最主要的对话框之一，基本上，对工程的所有设置都在这个对话框进行。该对话框中的内容，光盘例子里面已经配置好了，基本不需要改动，用户可以熟悉一下该对话框的内容。

注意：用户需要检查一下 Remote 标签下面，对 Remote Device 的配置，请选择对应的仿真器类型，如下图是使用的 PowerIce 仿真器时选用 PowerIceArm9 选项图片。如图 7-5 所示。

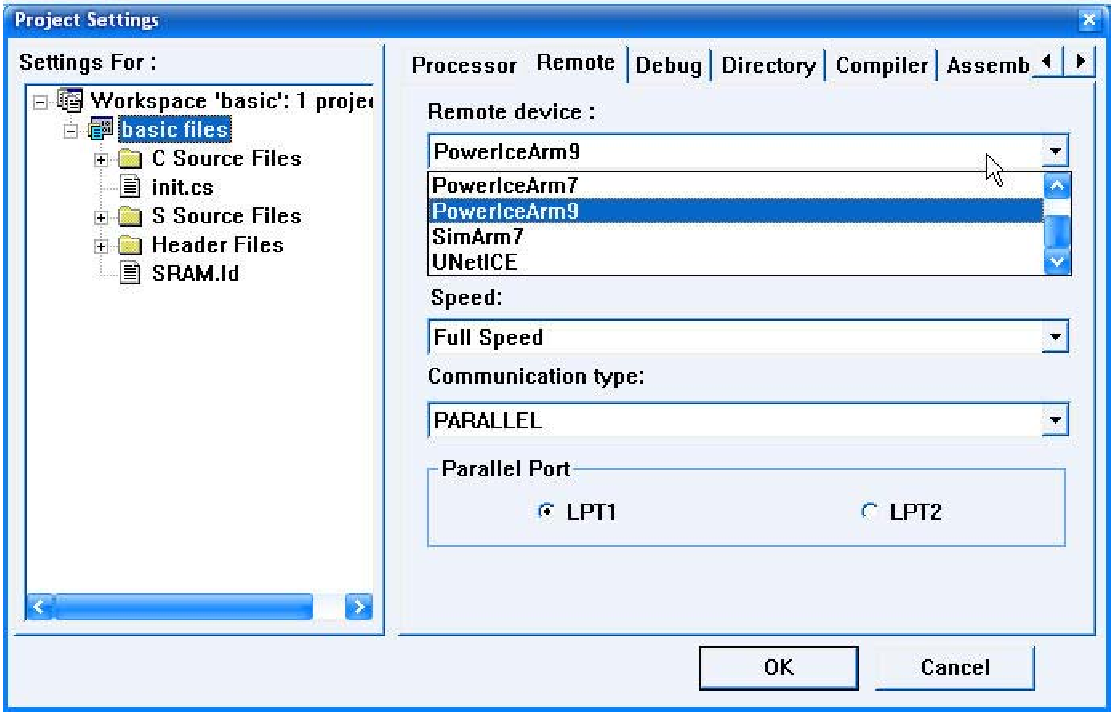

图 7-5

在 EmbestIDE Pro for ARM 中，点击 Debug-Remote Connect 菜单，或者通过热键 F8，连接评估板，应看到如下界面和提示信息。接下来，就可以通过 Debug 菜单，停止评估板，下载程序和全速运行观察结果了，如图 7-6 所示。

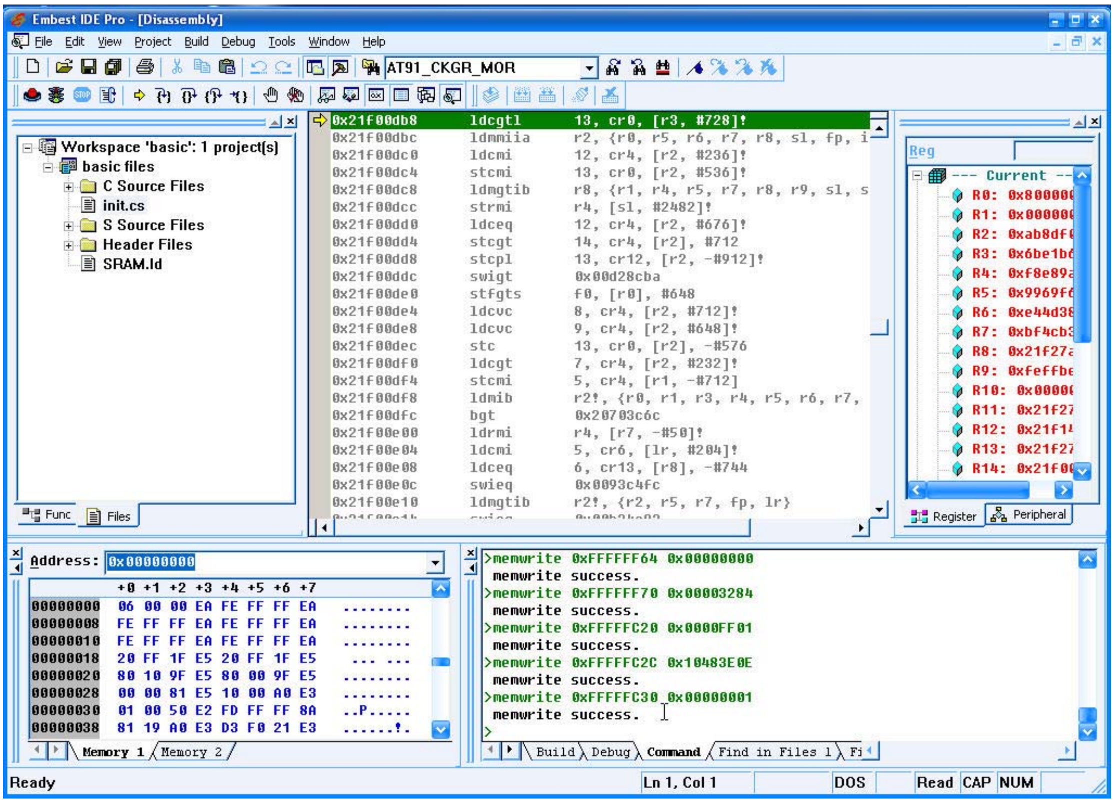

图 7-6

点击程序运行图标，程序运行结果如下，通过调试接口不停打印“.“号，如图 7-7 所示。

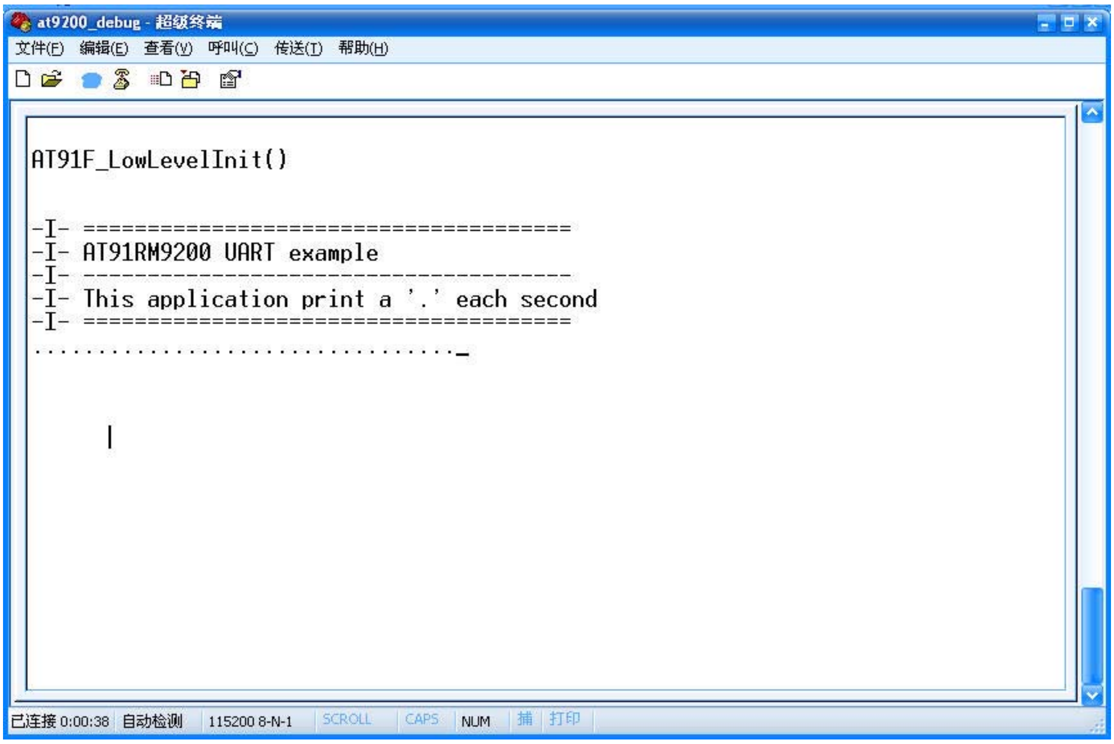

图 7-7

7.3 程序固化

通过上节调试，运行正常，就可以将结果通过 Embest 公司的 Flash 烧写工具写入到 Flash 中去。

首先将要烧写的编译生成的*.elf 文件转化为*.bin 类型文件，直接在 tools 菜单点击 Elf to Bin 栏即可。下图 7-8 所示。

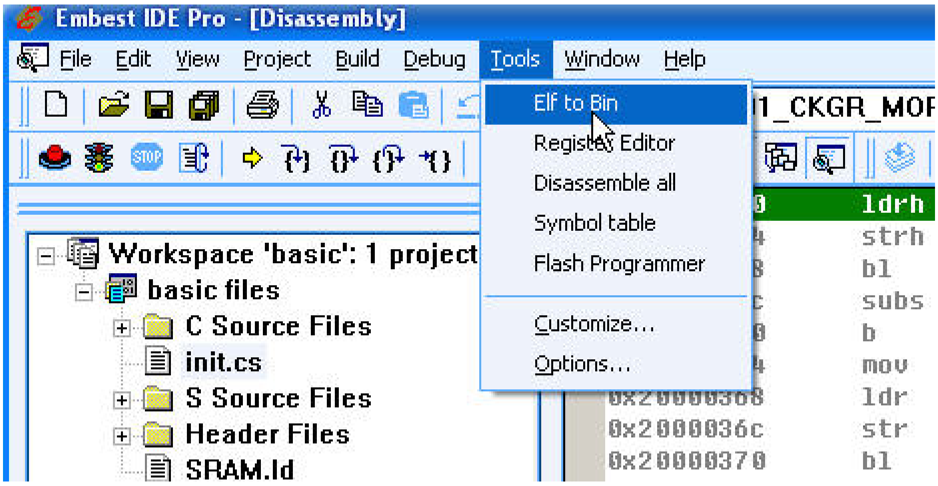

图 7-8

然后点击如上图所示 tools 下的 flash programmer，运行 Flash 烧写工具。运行结果如下图 7-9 所示。

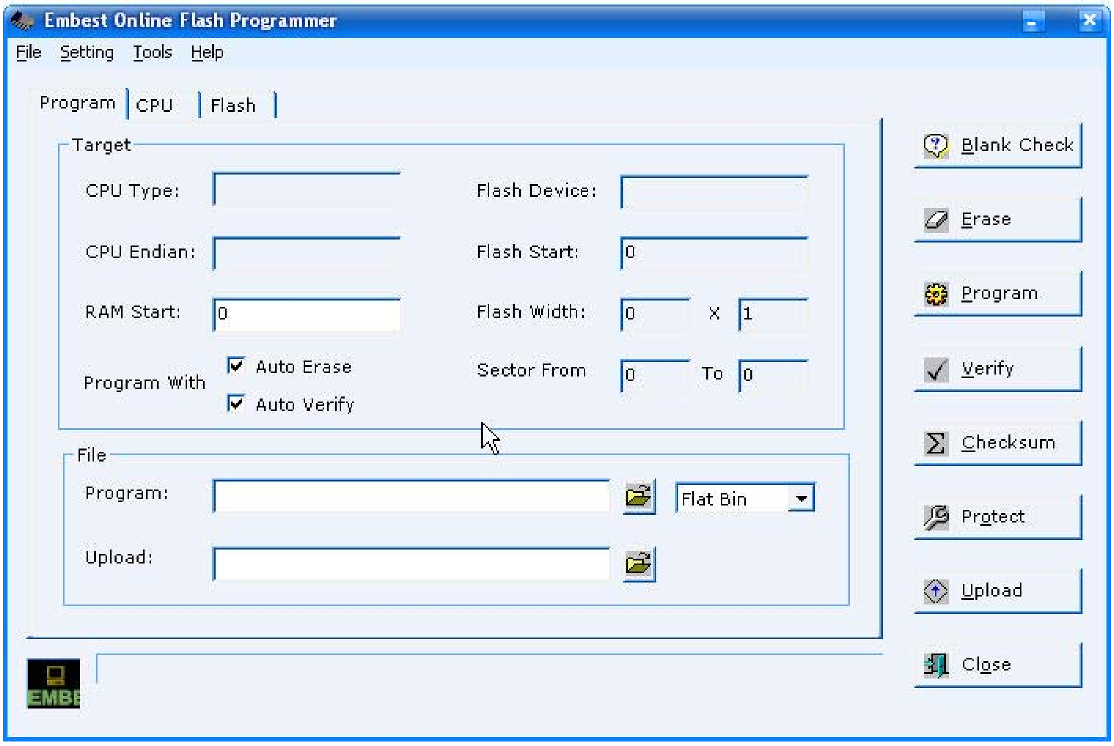

图 7-9

通过 File 菜单，选择 Open 相，打开 ATEB9200 开发板的 Flash 烧写配置文件，此文件在光盘的\tools\config file 文件夹下提供。如下图 7-10 所示。

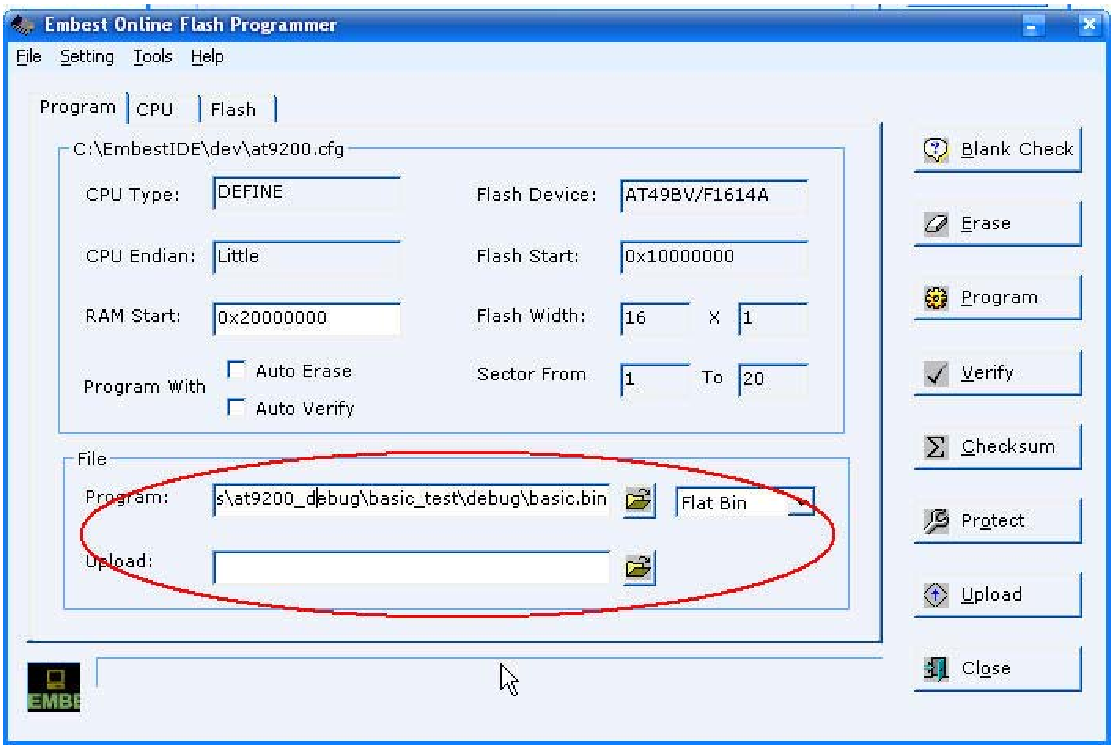

图 7-10

在如上图标记的 File 框的 program 栏中选择要烧写的烧写的文件,然后点击右边的工具条 Program 按钮，开始烧写，如下图 7-11 所示。

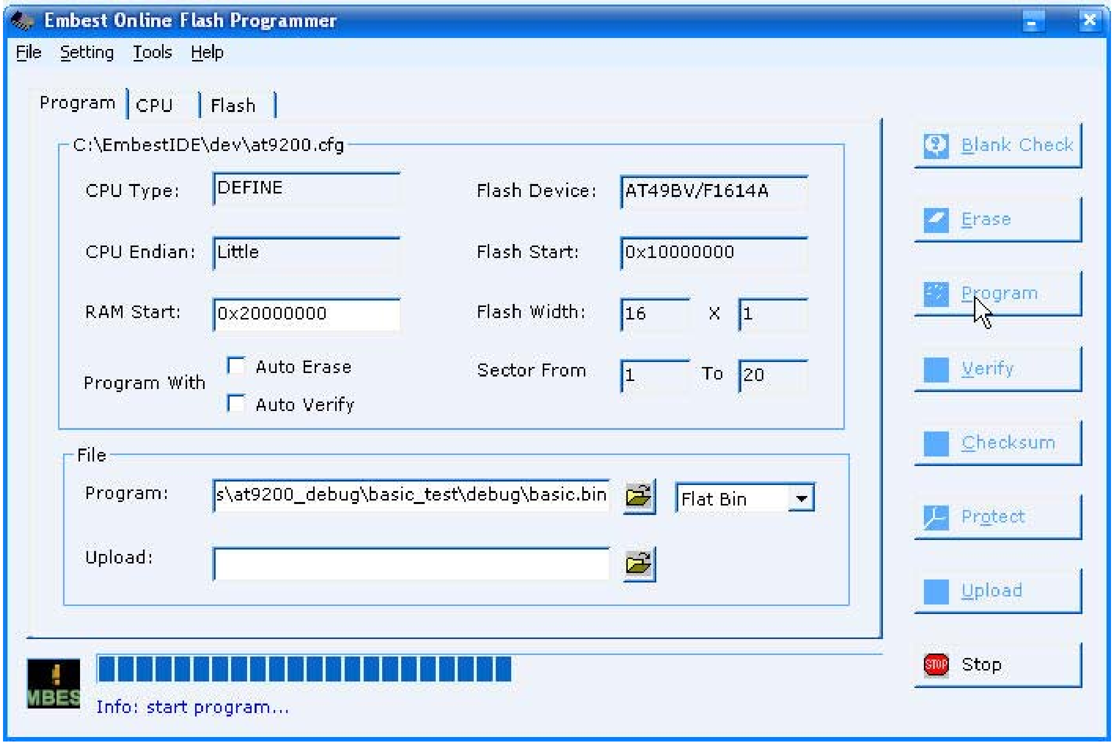

图 7-11

最后烧写完后提示：Info: program completed，表示烧写成功，如下图 7-12 所示。

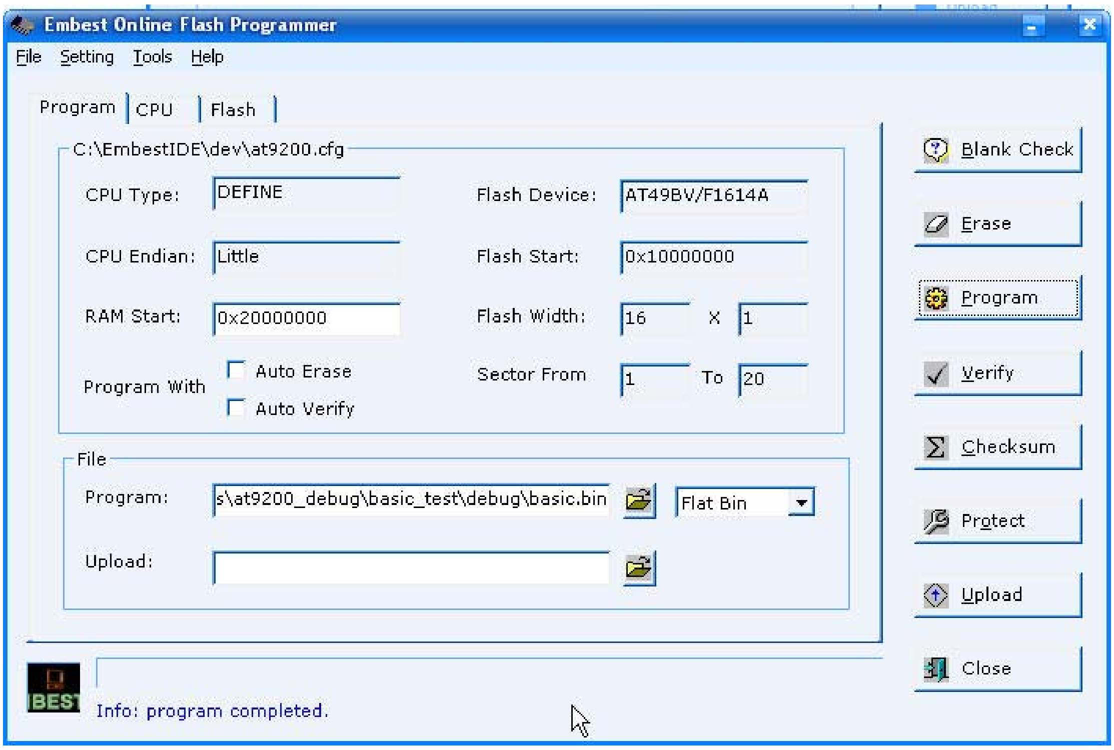

图 7-12

第八章 附录

8.1、跳线说明

跳线名称	跳线说明
SW201	连接时 DataFlash 使能
SW801	连接从外部存储器启动，不连接时启动内部 ROM 的 BootLoader
SW402	1-2 连接使能 JTAG 接口，2-3 禁止 JTAG 接口
SW401	SM 卡写保护，连接时写使能

第九章 Embest ATEB9200 系统售后服务及技术支持

深圳市英蓓特信息技术有限公司承诺为我们的客户提供相关技术支持。如果您在使用我公司产品的时候，遇到任何问题，可以通过下列途径与我们客户服务部的技术支持工程师联系：

英蓓特公司网站：

关于英蓓特公司产品的最新最准确的信息（包括公司产品信息以及相关资料），您可以通过以下网址得到：http://www.embedinfo.com。

技术论坛：

英蓓特公司提供两个主力论坛供我们的广大客户以及业界的工程师相互交流和学习。

ARM 开发论坛：讨论 ARM 技术、ARM 系列芯片、ARM 开发工具、ARM 嵌入式处理器的开发、ARM 应用的论坛。

Embest IDE 用户论坛： EMBEST IDE 用户技术交流、技术支持的论坛。

邮件：

用户可以通过邮件地址 support@embedinfo.com 直接与我们的客户服务工程师联系。

电话：

用户可以在工作时间拨打我们的客户服务热线电话 **86-755-25631365**。

警告：

请务必注意静电的防护。超过任何最大承受值，均会对产品产生永久损害。同时，不推荐在临界状态使用产品。

第十章 免责声明

本光盘提供的程序源代码、软件、资料文档等，深圳英蓓特信息技术有限公司不提供任何类型的担保。不论是明确的，还是隐含的。包括但不限于合适特定用途的保证。全部的风险，由使用者来承担。如果程序出现缺陷，使用者承担所有必要的服务、修改和改正的费用。